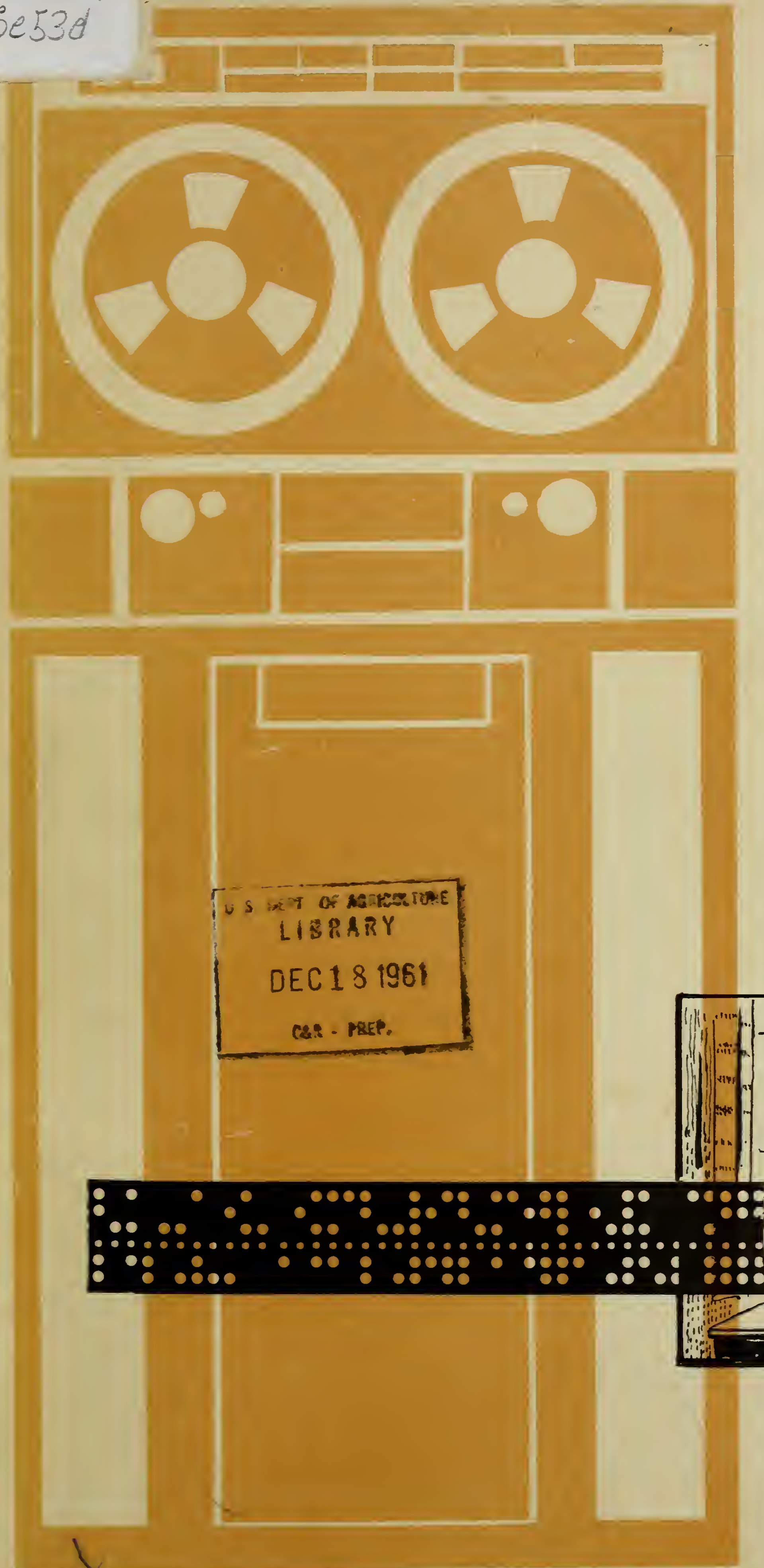


AUTOMATIC DATA PROCESSING SEMINAR FOR FEDERAL EXECUTIVES

UNITED STATES DEPARTMENT OF AGRICULTURE | GRADUATE SCHOOL

"ADP" is a ... potential in the realms of s ... ent is tremendous. The ... y both in government and ... my.

Perceiving t ... gent utilization, the Graduate ... series of ADP Seminars for Federal Executives. ... gh this seminar report, we seek to extend even further the understanding and appreciation of this new technology.

To those who devote themselves to the public service, we earnestly offer this volume. We hope you may find here both fundamental knowledge and intellectual stimulus to capitalize on the bright promise which ADP offers.

John Holden
Director
Graduate School

September 1961

Additional copies for sale by

THE GRADUATE SCHOOL
U. S. DEPARTMENT OF AGRICULTURE
Washington 25, D. C.

Price: $4.50 each

Foreword

This volume represents a stage in the development of a series of Seminars on Data Processing for Federal Executives designed to provide guidance to top level Federal Executives to assist them in evaluating the place of Automatic Data Processing in the work of their respective agencies.

The Purpose of the Seminars is as follows:

> "This seminar is intended to acquaint top Federal Executives with the scope, challenge, and capabilities of modern data processing techniques and equipment. It is designed to provide knowledge about what is involved in the use of these new tools of management and how they may help to accomplish the objectives and programs of Federal agencies and departments. Specifically it is hoped that executives participating in these Seminars will gain insight as to applicability, potential usage, effective utilization, and evaluation of this type of equipment, and management and organization problems."

As an outgrowth of some informal discussions with several members of the Department, it appeared desirable to appoint a Planning Committee to consider whether the Graduate School might be able to help fill a growing need for information in this field that has been developing with explosive rapidity.

The Committee concluded that a Seminar on Data Processing -- aimed at government executives at the higher levels -- could provide some of the guidance needed if it could successfully tap sources of broad experience and information both in and out of the government. It developed such a program, seeking to concentrate upon the major management problems involved, and to subordinate details and minor points except as they might be necessary to give reality and concreteness to the development of the subject matter. The Committee proposals were then reviewed by a number of advisors with extensive experience in this field. The Committee members and the Advisors are listed on the following page. Some modifications in the program were made as a consequence of this review.

The Seminar was announced in the fall of 1959 by an invitation to Federal Agencies to designate members of their staffs for enrollment. The demand for the Seminar announced in the fall of 1959 required its repetition in the winter-spring of 1959-60, and again (several sections) in 1960-61. The Seminars are being continued in 1961-62.

As a means of capturing in permanent form the essence of the material presented, the papers from the third Seminar (September 28, 1960-November 10, 1960) (with a few additions and substitutions for the purpose of rounding out the coverage by drawing on presentations at some of the other Seminars) are reproduced herewith.

The Graduate School is deeply grateful for the cooperation of those who have contributed to the success of the Seminars. To the Planning Committee, the Advisors, and to Earl E. Houseman, W. Henry Hill, and Carl B. Barnes who have served as Moderators, and to the speakers who have taken time from their busy schedules to share their experience and knowledge with the Seminar participants goes a full measure of appreciation. I want also to thank Edmund N. Fulker, Assistant Director, for his capable assistance at all stages of the program.

I want to express my appreciation also to Philip V. Fleming for technical editorial assistance, to Ben Murow for the art work, and to Mrs. Dorothy Washington, Mrs. Arline Saez, Mrs. Louise Hopkins, and Mrs. Barbara C. Pollard for their painstaking typing of the publication.

B. Ralph Stauber
Editor

GRADUATE SCHOOL PLANNING COMMITTEE

B. Ralph Stauber, Chairman, Chief, Agricultural Price Statistics Branch, Statistical Reporting Service, Department of Agriculture

John C. Cooper, Deputy Director, Office of Budget and Finance, Department of Agriculture

W. Henry Hill, Director, Navy Management Office, Data Processing Systems Division, Navy Department

Earl E. Houseman, Director, Statistical Standards Division, Statistical Reporting Service, Department of Agriculture

Charles F. Kiefer, Executive Director, Management Operations Staff, Agricultural Economics, Department of Agriculture

James W. Purvis, Management Analyst, Office of Administrative Management, Department of Agriculture

Edmund N. Fulker, Assistant Director, Graduate School, Department of Agriculture

GRADUATE SCHOOL ADVISORS

John Provan, Assistant Administrator for Administration, Veterans Administration

William A. Gill, Management Analyst, Office of Management and Organization, Bureau of the Budget

Charles A. Phillips, Director, Data Systems Research Staff, Office of the Assistant Secretary of Defense, Comptroller, Defense Department.

Joseph Daly, Chief Mathematician, Bureau of the Census.

John Roth, Director, Federal Incentive Awards Program, Civil Service Commission

William Orchard-Hays, C-E-I-R, Inc., Arlington, Virginia

Contents

PART 1

GENERAL INTRODUCTION

ADP IN FEDERAL AGENCIES, I.

Charles A. Phillips

Charles A. Phillips, Director of the Data Systems Research Division, Office of the Assistant Secretary of Defense (Comptroller), has principal staff responsibility for Review and approval of automatic data processing systems as applied to business type operations in the Department and general direction of the Department's ADPS program. With over 25 years with the Federal Government in the Departments of Defense, State, and Treasury, he is also a faculty member of the American University.

My assignment this morning is to tell you something about how we are using automatic data processing systems in the Federal Government. My remarks will be confined to <u>what</u> the equipment does for us in the Federal Service, as distinguished from <u>how</u> it operates. I propose not to deal in the abstract any more than necessary but to be as specific as possible in describing how we are actually using, or planning to use, ADPS. My talk will be divided into three major parts:

First, I would like to comment rather briefly on the general climate into which automatic data processing systems were born in the early 1950's; this is intended to provide a background for a better understanding of the tremendous interest in electronic computers and the phenomenal growth of ADPS in the short span of approximately ten years.

Secondly, I propose to give you a general feeling for the number of ADP installations in the Federal Government and something of the costs that are involved in their operation.

The first two parts will, in effect, represent a prelude to the main part of my talk, the meat of the coconut, on where and how we use ADP in the Federal Government. Because of the disparity in size, as well as the need for dividing the subject into manageable segments, I have grouped the ADP installations into two groups for purposes of this discussion: first, Civilian Activities, and second, Military Activities. I have a mixture of slides that were put together for other purposes, but will serve to illustrate certain points. (The written transcription has been edited to include significant figures from the slides that are needed for clarification.)

I do not intend to talk about factory automation, but to help to establish some perspective concerning the background, or climate, in which ADP was born, it will be helpful to consider first, the rise in productivity over the past half-century, and second, the changes in man-hours associated with this productivity. We have had a

tremendous increase in the production capability of this country in the past fifty years, with a particularly sharp rise since 1950. When we relate the manufacturing output to the man-hours worked, we find that with 9% fewer man hours worked in 1958, we are actually producing 7% more than we did in 1953, just five years before. This rise is generally attributed to the fact that within the past nine or ten years there has been a tremendous increase in the automation of factory methods. Next it is instructive to consider the effect that automation has had on personnel levels within the past fifty years. Within this time span, our production in the U. S. has increased by approximately 700%, but during this same period our factory workers only increased by approximately 100%. Compare this with the 400% increase in office workers during the same period - in other words, we have added four office workers to our employment rolls for each new factory worker. This, I think, is indicative of a rather general failure in this country to automate our paper work to the same degree that we have automated our production processes.

We should keep in mind that during the fifty year period from 1910 to 1960, American business was characterized by a big increase in paper-type operations, such as insurance. From a relatively minor position fifty years ago, insurance has grown to the point where, in the life insurance field alone, there were over 270 million policies in force last year, each with a dozen or so bookkeeping entries per year - a gigantic paper processing operation in itself. Bank operations and the general use of checks are also major contributions to this high rate of growth in paper-work. Indicative of this rise, it is estimated that the use of checks has grown from approximately eight billion checks per year in 1952 to over thirteen billion in 1960 and it is predicted that this will further increase to twenty-two billion checks per year by 1970. With each check being handled from ten to twenty times in its business life, this means from 130 billion to 260 billion handlings of checks or check transactions in 1960. And don't forget the related increase in deposit slips, bank statements, and ledgers. To keep up with this soaring paper work, we added over 100,000 additional office workers in this country last year.

Let's next consider the trend in the number of office workers in relation to factory workers. From a relationship of 1 office worker to every 20 factory workers fifty years ago, we now have 4 office workers to every 20 factory workers. The clerical force almost doubled within the past fifteen years and we may soon have more clerical workers than we have factory workers. There are now over eight million clerical workers in the U. S. -- enough to populate a city the size of New York.

The government is responsible to some degree for this tremendous increase in paper work. Last year there were over 350 million tax forms filed. Social Security reports, which are required four times per year, were submitted on over 120 million active accounts. A few years ago,

the Hoover Commission reported that the Federal Government produced about 25 billion pieces of paper each year. This is enough paper end-to-end to reach to the Moon 13 times. While the Russians were shooting at it, we were producing enough paper to pave the way to the Moon four times, and in triplicate, of course. We are rapidly becoming an economy of "paper-shufflers".

Back in 1950 when they were contemplating entry into the field of manufacturing automatic data processing equipment, the Radio Corporation of America made a market survey of the paper-processing operations in this country. Through this survey they found that government and private industry was spending around $32 billion dollars a year for clerical personnel plus an additional $2 billion for clerical tools. Here the tools I am talking about are non-automatic and include pencils, paper, adding machines, and other non-automatic clerical equipment.

In my opinion, there are three principal reasons for the tremendous increase in office workers. Two of these reasons I have touched upon briefly: (1) the general increase in production, through improved technology and the use of automated methods, and (2) our very great (and getting ever greater) government programs with their heavy reporting requirements. The third reason, which might be the most important, is the complete failure on the part of the office equipment industry to bring forth any basically new tools for a time span of about fifty years. Punch-card methods, which were invented in 1890, were the last new concept before the advent of the electronic computer. There were many improvements in punch-card methods, but no basic new concept such as we now have in the electronic computer, which could help us cope with the rising tide of paper work.

You may find it of interest that in 1959, with electronic computers almost ten years old, and with accelerated office automation efforts in the last few years, only 20% of our paper work has been mechanized or automated. There still remains about 80% of our paper work that is being accomplished through completely hand methods - without any mechanical aid whatever. Is it any wonder that the number of computer manufacturers is growing rapidly?

Many of you probably read the October, 1959 FORTUNE article on the computer business in which it was estimated that computer manufacturers will gamble over $500 million on this market in the next few years. I think this is conservative since I was told by one company that they are prepared to invest over $150 million to establish a strong position in the industry. The FORTUNE article speculates on the relative standing of companies, and predicts that only four out of the nine principal contenders will survive in this market. There is no question that the business is getting highly competitive and at the same time growing like a mushroom. An article on data processing in the July 4, 1960 issue of NEWSWEEK predicts that the computer industry will grow to about

$1.5 billion in sales, service, repairs, and programming in 1960 and that it will swell to a $20 billion per year industry in the next ten years. The General Electric Company, a late starter in this field but an important one, announced in September their intention to open eleven data processing centers in major cities over the country together with ten sales offices. GE also estimated that their computer deliveries will increase 33% in the next two years.

With this as a background on the recent and present climate, let's now look at how the Federal Government is using, or planning to use, automatic data processing systems. First as to number of equipments installed and cost of operation:

Based upon a General Accounting Office survey of two years ago, which I have updated to September, 1959, we find there were sixteen different civilian agencies, out of a total of fifty-two agencies, that were using ADP. These 16 agencies had a total of 61 ADP installations at 51 different locations. A recent inventory by the Bureau of the Budget indicates that by the end of Fiscal Year 1961 there will be 23 civilian agencies using 152 different ADP sytems at 128 locations.

The military departments are reported on separately since they have about 70% of the total for the entire Federal Government. As of September 1959 the Department of Defense had a total of 143 ADP systems at 108 different locations, several of them overseas. Based upon the Bureau of the Budget report, the Department of Defense will have 458 ADP systems installed at 288 different locations by the end of FY 1961.

Again using the Bureau of the Budget report as the source, we find that in Fiscal Year 1961, the Civilian agencies are estimated to spend at the rate of $120 million per year for operating costs with an additional $5.1 million of capital outlay in the data processing field. The Department of Defense is estimated to spend at the rate of $225 million for operating costs and $9.5 million capital outlay for ADP in FY 1961. The figures for capital outlay include the costs of preparing the site for computer installations together with the purchase of some systems. Most ADP systems are rented and the rental costs in both civilian and military agencies are about one-half the total operating costs. If we combine the civilian and military figures, we find that the total operating costs for the Federal Government are approximately $345 million plus an additional $14.6 million of capital outlay. It should also be mentioned that the figures for the military agencies do not include any computers that are part of a weapons system, used for operational purposes, or in a classified project.

Now that we have an idea as to the number of ADP systems in the Federal Government and their cost, let's see how they are being used. Quite obviously I can't cover all the applications on the 600 or more ADP systems in the time alloted. For this talk I have identified and

listed 20 different applications of ADP in civilian agencies to which I will refer very generally and quite briefly. I have a similar list of approximately 14 applications in the military department which I will describe briefly and generally. Following this, I have selected two civilian and two military applications to describe in some detail. In the general listings I will start with the older and more general uses, proceeding to the newer and more limited types although I ask you to recognize that the order itself is only an approximation.

The earliest and I believe the widest use of computers in the civilian agencies is for engineering and scientific calculation. Since this type of application is well understood and will be discussed again in describing the military applications, let's move on to the second most general use: the processing and tabulating of statistical data. This refers to operations such as those performed by the Bureau of the Census, which was the first user in the Federal Government on a non-scientific basis, and which deserves much more than a passing reference. The Bureau of the Census took delivery in April 1951 of the first UNIVAC, which was the first large scale commercial general purpose computer. Since delivery took place during the peak of the 1950 population census operations it had little impact on such operations, for input and output equipment was not very well developed at that point in time. In preparation for the 1960 population census the Bureau added another Univac plus two more of the later model UNIVAC SCIENTIFIC (model 1105) and arranged for the part time use of UNIVAC 1105's at the University of South Carolina and the Armour Research Foundation -- a total complement of six large scale computers. Also, during the intervening period, Census, together with the Bureau of Standards, had developed new automatic equipment for input to the computer, called FOSDIC (which is the subject of a separate discussion). With this total equipment, Census expects to complete the 1960 population census much quicker, at less cost and providing more information. During the 1950 population census the peak employment of the Bureau was approximately 9500 employees. On the 1960 census the peak employment is expected to be around 4000 -- in other words, with a population increase estimated at about 20%, the "big count" will be done with an employee reduction of about 45%. Other civilian agencies using ADP for statistical purposes include the Departments of Labor, Justice, Agriculture, and the Veterans' Administration.

There are approximately 30 civilian ADP installations, either doing or planning to do payroll operations. Along with payroll has been grouped the accounting for time, leave, and retirement, which is normally the scope of an application identified as "payroll," although the fact that an agency has payroll on ADP doesn't necessarily mean that leave and retirement accounting are included under the same system. The largest of the civilian payroll operations is in the Post Office Department, which I understand is now payrolling approximately 500,000 employees every two weeks, with only a seven-day pay lag, which compares very favorably with the ten-day pay lag found in most agencies. I might add that the payroll job is probably the most popular one in private industry.

The check payment and reconciliation application of the Treasury Department and the General Accounting Office is a very interesting one that I have selected for a more detailed description which I will cover later.

The wage record maintenance and claim computation application of the Social Security Administration has been described as the biggest bookkeeping job in the world. Social Security has over 170 million accounts with more than 120 million on an active basis and requiring approximately 4.4 postings per year on the average. Social Security accounting has been on a mechanized basis since inception, and automatic data processing was originally installed to avoid setting up a second card file to accomodate additional data on around 117 million accounts. The first application was in the maintenance of wage records which are now updated quarterly instead of annually under the previous system. The Assistant Director of Social Security estimated that the wage record application alone would save from $2 to $5 million annually. An additional application recently converted is the preparation of abstracts of earnings for persons who have filed claims or requested a statement of earnings. Social Security has pioneered in the development of methods by which employers report the earnings of employees in machine language through the submission of reports in the form of punched cards or magnetic tape. For example, the Army Finance Center now transmits information on the earnings of Army military personnel to Social Security on magnetic tape. It is interesting to note the impact that ADP has had in the personnel strength of Social Security. The initial application of maintaining wage records prevented a personnel increase of from 250 to 300 that would have been required under previous methods. It is estimated that the Social Security Bureau would require from 700 to 800 more persons today without automatic data processing. It should also be noted that this change has been accomplished without any actual separation actions and entirely through turnover and attrition.

Property and inventory accounting - both in items and dollars - is quite a popular application in the civilian agencies. Most agencies which have this application and the payroll application also do appropriation and fund accounting on ADPS. When the basic data on personnel, supply, or contractual costs are reduced to machine-sensible form by preparing a punched card, punched paper-tape, or similar machine input media, it makes sense to extend the ADP operations to appropriation and fund accounting and cost accounting. To my knowledge there has been no instance in which appropriation, fund, allotment, or cost accounting has been placed on ADP unless there was an underlying or basic application for property and inventory accounting or payroll.

Now I would like quickly to review several additional types of applications that are not in general use or in the early developmental stages. The Atomic Energy Commission (which incidentally has 10% of the computers in the Federal Government, mostly on scientific calculations)

uses ADP for production control purposes. The Housing and Home Finance Agency uses it to maintain loan accounts. The Post Office is using ADP to process claims for air mail transportation faster than ever before. Internal Revenue is using computers to develop statistics and to calculate tax refund computations and in addition has a very ambitious program to extend ADP into most of the tax collection operations. Civil Service is using it for actuarial work;the Department of Agriculture uses ADP to account for the price support, soil bank, and wool programs. Several agencies are exploring the potentials of the equipment or beginning to use it for data storage and retrieval - a special application of which would be searching patent applications.

The Weather Bureau has come to rely on computers for numerical weather predictions and is moving toward this as a dominant technique. The Federal Aviation Agency has a very ambitious program centered around the use of ADP for processing flight process data as an effective method of air traffic control - a subject of interest to all of us. And we must not forget that in the race for space, satellite meterology and its related sciences are almost completely dependent upon computers. The potentials of these last three applications which are directed toward problems involving the basic missions of the agencies, as distinguished from the administrative functions such as payroll and supply accounting, may very well have a more profound impact upon our future history than all other civilian agency applications combined.

Next, let't look at the general range of ADP applications in the Department of Defense. The earliest, and probably still the most general use of computers in the Department of Defense, is in scientific and engineering calculations. Without doubt, the first use was in ballastic calculations at the ballistics research laboratory of the Army at Aberdeen where the first electronic digital computer was born. Until very recently, the Naval Ordnance Research Laboratory at Dahlgren, Virginia had the fastest electronic computer, the NORC. To mention a few others, the Air Research and Development Command at Dayton, Ohio; the Navy installation at Point Mugu; the missile ranges at White Sands, Cape Canaveral, and Huntsville, all have major installations of computers applied to scientific, research, and development problems. In the same general category are the computers at Navy's David Taylor Model basin, which will soon accept the second copy of the LARC, one of the largest and fastest computers yet developed, and the Army Map Service which uses the UNIVAC I (a very early serial number) to plot map coordinates. In fairness I should mention that the Map Service also has plans to up-date its system.

The second most general use of ADP in Defense is in supply and inventory control - which should probably include the cataloging function. In DOD this type of application represents about 80% to 85% of all our business-type applications. Although requirements determination might technically be considered as a different application from supply and inventory control, it falls within the same category.

Technical failure data analysis is an especially valuable tool in the Department of Defense. This application provides data essential to the effective development of our weapons systems as to the reasons for failure of critical components such as electronic gear. Such analysis will pinpoint the failure by reason, such as temperature, humidity, environmental condition,etc., and identify the failing component by date of acquisition, manufacturer, or other index needed for developmental efforts.

Defense has a number of ADP applications doing real and personal property accounting and control. For example, the Air Force maintains, at the Oklahoma City Air Material Area on a large-scale ADP, a world-wide inventory of jet engines. This inventory of approximately 120,000 jet engines worth around $200,000 each, is updated daily on the basis of change in the status of the engine; that is, whether in-service, awaiting repair, undergoing repair, in storage, etc. At $200,000 apiece, this inventory represents a substantial amount of money and the Air Force has been able to reduce the inventory of engines by approximately one-third. This reduction was due in part to the ability to know exactly where each engine was and its condition.

Military personnel accounting and reporting is another large ADP application in Defense. This application does not encompass payrolling. Instead it is designed for personnel management which needs to know where the officers and enlisted men are assigned, for how long, and other data such as military speciality, rank, age in grade, education, dependents, and other information needed to manage military manpower effectively. Current and accurate data of this nature help us hold our military personnel pipe-line requirements at a minimum. Actually the pipe-line requirements have been reduced quite substantially in the past few years.

Civilian payroll preparation has generally been left on punchcard equipment or conventional bookkeeping machines in the Department of Defense - in fact, I don't know of a single instance in which payroll was the principal application. We do have payroll on ADPS where there are large concentrations of our civilian forces such as the departmental offices in Washington, or large industrial activities such as shipyards, arsenals, or overhaul and repair depots. The government payroll, we find, is comparatively simple, and it hasn't been too good an application except for the larger groups, or where the basic data needed for payroll were put into machine-sensible language for another purpose.

General accounting and cost accounting is done on the computer only where there are other applications utilizing the same basic data. In other words, if the activity has a supply or inventory application and a payroll application on the computer, it is logical also to do the general accounting and cost accounting on the same equipment.

War gaming and target analysis relate to studies of individual weapons and their projected effectiveness against specific targets. Mobilization planning is more general and includes post attack damage assessment. In both these applications the computers' high speed and tremendous storage capacity permit the extension of such studies far beyond what was possible with previous equipment.

A few words about the last item on the list - language translation. A good part of the work in this field is classified, but some very interesting developments were made public in May 1960 when equipment developed by IBM under a contract with the Air Force was demonstrated. This new equipment has been translating the Russian newspaper PRAVDA into rough but meaningful English since June 1959. In this application, a Russian dictionary of about 55,000 word stems has been compiled and stored in the machine along with about the same number of word endings, which together correspond with around one-half million words in the text. It is expected that the dictionary will ultimately contain about 400,000 word stems. The equipment is now capable of translating at the rate of approximately 30 words per second, being restricted in its speed by paper tape input and electric typewriter output. The heart of the system is a new device called a "photoscopic disc memory" on which the Russian input words are matched to Russian dictionary words and the English language equivalents printed out. Input devices under development are expected to feed the computer at the rate of 40 words a second. The grammatical capacity of the translator is well below college level but it is expected to improve.

The importance of this development in language translation may be recognized in the fact that less than one percent of the world's foreign technical literature is now being translated into English. In the Soviet Union, there are over 2600 full time language translators, plus about 26,000 part time persons who collectively translate and publish Russian language translations totalling about one-half million abstracts of translated books and articles. Using conventional methods of translation, which cost from 8 to 10 cents per word, the United States is not apt to catch up in the translation race. Computer translation may provide the answer here.

This completes the general listings of applications in the civilian and military agencies, and I would now like to describe in a little more detail four specific installations. I have chosen two in operation and two in the planning or early developmental stages, evenly divided between the civilian agencies and the military agencies.

The first specific application I will describe is the Treasury check reconciliation job. Everyone is familiar with the function of reconciling a checking account and the Treasury job is essentially the same, except for the magnitude. There are 2300 civilian and military disbursing officers drawing checks against the Treasurer and they write over

350 million checks per year. This represents about one million three hundred thousand checks per working day which, on punch-card stock (and all government checks are now on punch-cards) would make a stack as high as the Washington monument. I thought this check volume was probably the biggest in the country but recently learned of a commercial application that is as big, or bigger. The Chase Manhattan Bank of New York is reported to process one and one-half million checks per day for approximately $1.25 billion and I understand that this check processing operation requires 2000 persons.

The Treasury check reconciliation application was undertaken as a joint improvement program by the Bureau of the Budget, the General Accounting Office, and the Treasury, together with the operating agencies. The feasibility study and the application design phases of this job were well and carefully done over a period of almost three years prior to actual installation. In my opinion this was one of the best jobs of planning in the entire Federal Government and will compare favorably with any in private industry.

If we look at the "before" and "after" statistics on the Treasury job we find that it affected both the Treasury and the General Accounting Office. In fact the GAO Reconciliation Branch was reduced from 384 to 15 employees while the Treasury check payment and reconciliation division was reduced from 371 to 255 employees. As evidence of the good planning and personnel action, the total reduction of 485 positions in both agencies was accomplished almost entirely by attrition. I might add that this is one of the few ADP installations where you will find such a dramatic personnel reduction.

Let's next look at what was done under the new system and how it affected the twelve Federal Reserve Banks and their twenty-four Branches that are involved in the receipt and payment of Treasury checks. Under the previous system, the bank which first received the check from the commercial bank handling it for collection would sort and list the checks, balance them, review them, and then forward them to the designated paying bank where the same sorting, listing, and reviewing would be done again, after which the checks would be forwarded to the General Accounting Office. Under the new system, the Federal Reserve Bank or Branch Bank first receiving the check from the commercial bank simply lists the checks in the order in which they are received and forwards them unsorted to the Treasury Department. I should mention that under the previous system, the GAO would manually sort the checks in a single sequence and then manually check them against the listings of check issues submitted by the various disbursing officers. Under the new system, the checks are not physically sorted at all, but are stored in the same random order in which they were first presented for payment in the commercial bank and submitted by such bank to the Federal Reserve system.

Without going into detail on the application itself, let me describe some of the results. I mentioned the elimination of physical sorting of checks before storage, which is probably responsible for the greatest personnel reduction. The new system includes an automatic search for "stop-payments" as one of the first actions. There is also a detailed comparison of check payments against check issues which has resulted in the prompt detection of counterfeiting or other irregularities. There is an automatic listing of outstanding checks and reconciliation of the disbursing officers' checking accounts. Only checks in excess of $1000 are visually examined, which means that only 2% of all checks issued receive this scrutiny. The new system has also leveled the work load and largely eliminated peak-load problems.

Through this new scheme of using data processing equipment in the check payment and reconciliation processes there have been some substantial cost reductions. I mentioned the reduced work in the Federal Reserve Banks and their Branches, which represents a saving of approximately $1.3 million annually. Savings through personnel reductions were, of course, greater in the GAO, and the overall reduction of 485 positions is offset from a saving standpoint by the additional costs for equipment rental. From an overall standpoint and considering the costs and savings in both agencies, there is a net cost reduction exceeding $1.5 million per year. This has recently been verified in a GAO report and continues to be reported by the Treasury each year as a saving. From a cost reduction standpoint, the Treasury check reconciliation job is an outstanding example, but there aren't too many large-scale operations that lend themselves to automation as well as the Treasury job.

Next I would like to turn to the military and tell you about an application in the Navy, at Mechanicsburg, Pennsylvania. In Navy vernacular, the Ships Parts Control Center at Mechanicsburg is the "supply-demand control point" for ships parts. At Mechanicsburg they prepare allowance lists for each specific ship of the fleet which shows all of the components and the spare parts for that particular ship. Such allowance lists are used by the ships as a reference for maintenance and for the ordering of components or spare parts required in maintenance. They also prepare a cross-reference stock number sequence listing which shows the various uses for each given part. They maintain technical files for cataloging and prepare load lists for supply ships in addition to the main job of inventory control and requirements determination covering approximately 5 million spare parts and 70 thousand major components with an inventory value of approximately $495 million.

When Mechanicsburg made a feasibility study approximately two years before the installation of ADPS, they identified the following problems under the old system: They were maintaining a punch-card file of over 34 million cards, a substantial job. Their stock status report, prepared on a quarterly basis for all items, wasn't sufficiently timely for good management action. Because of the high volume, they were unable to

do a careful analysis as frequently as they thought necessary. The technical record files were not current; neither were their allowance lists or load lists. They also felt a definite need for more meaningful data for budget development.

Time will not permit a description of the various applications but here are several salient points under their new ADPS system which has been in operation about two years: Their inventory files are now on magnetic tape and on a completely current basis. Early in the application they updated the files weekly but they found this to be more frequent than necessary and changed to a basis where they update the file every second week. They are now determining their requirements position on a monthly position, although at the outset they did this every second week, which was found to be too frequent. They have devised a system under which they print out only those items which require management action. They are calculating safety levels and economic purchase quantities for each item instead of following rough general averages by category. Their technical files are not updated monthly and they are able to prepare their allowance lists and load lists whenever required. They are also developing much better data on which to prepare budgets.

Mechanicsburg is actually using the principle of management by exception which, I think, is honored more by its exception than by its observance. They have a much better mobilization planning capability and have cut the procurement lead time by about 30 days in administrative processing. They have done some consolidation of allowance lists and otherwise improved them with resulting improvement of supply operations aboard ship. They have actually eliminated 67 positions in the data processing operations. This did not result in a reduction of 67 jobs -- the billets were transferred to other functions within the supply complex, and the people went with the billets. They have cut the amount of time required for a complete analysis of their inventory from 17,000 hours to 80 hours. On the first 9000 items that were reordered after the ADP system was installed, they effected an inventory reduction of over one million dollars, such reduction being possible through the reduced lead time.

On the basis of the experience of the Treasury check reconciliation job and the Mechanicsburg application I have just described, we have a rather rosy picture and I hasten to caution you that it isn't always this good. Both Mechanicsburg and the Treasury did an unusually careful and thorough job of planning, but even with a competent job of planning, you don't always get such good results. Remember, there is nothing inherent about a computer that will assure its effective use, and we must be constantly vigilant to avoid the tendency to use its capabilities for unnecessary purposes. We have been making some careful reviews of our ADP installations in Defense and we find there is a general tendency to overestimate the benefits and to underestimate the costs. This doesn't mean that you won't necessarily realize the savings you anticipate, but

the targets may be a little bit unrealistic. Even with the careful planning in Treasury, it took a little longer than they expected, it cost a little more than they had planned, and the savings were a little less than they had hoped.

We also have a general tendency to attribute to data processing some benefits that are influenced by other factors. For example, I mentioned the jet engine management application of the Air Force at Oklahoma City. To complete the picture I should have at the same time told you that when we improved the information on the status of our jet engines, we also began to airlift jet engines; also that engine technology had substantially improved the life of jet engines. ADPS must certainly share the credit for inventory reduction with these two other important factors.

For another example of how the credit must be shared: The Quartermaster Depot at Richmond reduced their inventory by approximately $60 million within the first year after they introduced ADPS. When I brought this to the attention of our Budget Division in OSD and told them that ADP should be credited, at least partially, with this reduction, they said "No, we just cut their budget by $60 million." Maybe this is a "chicken or egg" type of sequence -- were they able to meet the budget cut through ADPS, or did the improved knowledge on inventory make the budget cut possible?

Next I would like to describe two additional applications which are still in the planning or early installation stages -- one in the civilian agencies and one from the military. In this last part of my talk I will first describe a planned installation of the Veterans' Administration in Chicago.

The Veterans' Administration has a monumental paper-work operation. To give you an idea of the size more than one-half million patients were admitted to VA hospitals last year and the average patient population was over 110,000; over $3 billion in benefits was paid to veterans and their dependents last year at the rate of $250 million per month; the VA life insurance program involves approximately 6.4 million policies with a face value of over $43 billion. These and other large scale operations convinced VA that ADPS offered advantages and benefits in its daily work. The application that I shall describe is the Death and Disability Compensation Program of VA, now being installed in Chicago.

In its rough dimensions, the job is to maintain some 4.7 million separate payee accounts on death or disability compensation and pensions; to advise the Treasury monthly of any changes in payments to such payees; to maintain necessary fiscal data and prepare required statistical reports; and last, but not least, to maintain a close contact and relationship with the individual veteran, wherever he may be. The objective is to improve this operation and, at the same time, to reduce the cost.

Here is the VA plan: They propose to establish at Chicago a central data processing center from which point they will maintain the benefit accounts which were previously kept at fifteen different locations. They will continue to maintain contact with the veteran at the local level -- I believe they have approximately 65 different state or city offices -- where they will adjudicate the claims, and then send the data by mail to the data processing center in Chicago. At the outset, data will be sent to Chicago in the form of punched paper tape, developed as a by-product in the preparation of the necessary claims forms at the local offices. The data processing center in Chicago will maintain the benefit accounts, process the monthly changes, and furnish the Treasury Disbursing Office with notice of such changes. The central office will also prepare all necessary fiscal and statistical reports on the program. Treasury will also centralize the preparation of benefit checks at the one office in Chicago, rather than at the twelve Regional Disbursing Offices. Advice as to changes in the payment file will be furnished the Treasury in the form of punched cards (although it is later planned that this information may be supplied through the media of magnetic tape when both offices can supply or utilize this form). The Treasury Disbursing Office in Chicago will prepare the checks and mail them to the Veteran or dependent.

One last but important step in the plan is for the Treasury to sort the checks in order by state or city before delivering them to the Chicago Post Office, thus making this operation unnecessary in the Post Office and speeding delivery of the checks. The Veterans' Administration hopes that this new system will result in a $3 million direct reduction in operating costs with improved service to the veterans. Treasury, through the consolidation of the check payment operation at a single office, expects to reduce the cost per check, but is not yet prepared to say what this reduction will be. Treasury will also jointly use with the Veterans' Administration certain records which previously were duplicated. The two agencies will be located in adjacent offices thus making possible this sharing of records.

The Post Office will also benefit substantially from the plan through the delivery from the Treasury of checks sorted by state and city. A sort of 4.7 million items per month represents a substantial workload that will be avoided. The military is also cooperating with the Post Office in a similar manner by sorting checks before mailing at the Air Force Finance Center in Denever, Colorado.

The last application that I will describe is the Air Force Ballistic Missile Support Program, which is now under installation at the San Bernadino Air Material Area. I am sure that everyone recognizes the importance of the Air Force Ballistic Missile program and the fact that it has the highest priority in Defense. Without question, this is the largest single military program ever undertaken by the Nation -- it had cost us well over $5 billion by the end of the last

fiscal year. Because of its tremendous costs and its importance to our national security, I'm sure everyone would agree that such a program must have (1) an extremely fast reaction time, (2) a state of constant readiness, and (3) a high degree of reliability.

While we can readily agree on the essential features of a missile force, there are relatively few who are aware of the complexity of the support problem. Some of the unclassified statistics on the Atlas missile will serve as an example: there are 400,000 parts on the "bird" alone; 500 components have time significance; 4,000 instruments need calibration; there are 8,000 relays in the launch control equipment; the automatic program check out equipment has 1,800 relays and 6,000 terminals; and there is enough wiring in the "bird" to circle the world 10 times. I don't have similar statistics on the Thor, Titan, or Minuteman programs, but one statistic on the Thor should be of interest: only 1.1% of the items on the Thor represent 60% of the total cost. Compare this with the fact that in jet aircraft, only 3% of the items represent 55% of the expenditures. Another thing that complicates matters is that in the Air Force Ballistic Missile program there are 25 major contractors, hundreds of subcontractors, and thousands of vendors and third-tier contractors. In all there are over 90,000 civilian and military personnel engaged directly or indirectly in this program.

A most important aspect of the ballistic missile logistic support program at San Bernadino is the complete integration of the various functional areas relating thereto. The objective is to integrate completely supply management with maintenance, transportation, personnel, and financial management. Vital to such a program is the speed with which the data can be made available to the manager of the support program. Such information must not be historical, it must be "during the fact" information, or even better -- "before the fact," if this kind of information can be made available.

Not only must the functions be integrated,but also the different physical locations and activities in the ballistic missile logistics complex. This means a high-speed communications network which will tie together the logistics support manager with the strategic air command and the ballistic missile division and will also link the ballistic missile support bases with the weapons systems storage sites, the depots, and the contractors. At the heart of such an integrated system is an automatic data processing system and switching control which will receive status information from all the different locations and activities in the complex; process such information and supply immediate answers to questions from these same customers. Such questions as: what needs repair -- what are the needs from current contracts, or from repair -- what stock do we have on hand -- is it enough, or too much -- what does the base need -- what does the depot need -- or why is the missile not ready?

The long range plans of the Air Force will extend this integrated system down to the missile base itself where the launching sites will be tied into a base ADP center and switching control which, in turn, will be connected with base supply, personnel, maintenance, and comptroller offices, and with the ballistic missile support center at San Bernadino. Developmental work on this concept is now under way at Vandenberg AFB, the first operational missile site.

This completes my prepared description of Government ADP installations, but I will be glad to answer any questions that I can on the subject. There are, of course, many excellent Government ADP installations that I have not mentioned because of time limitations. In my opinion, we have many ADP installations in Government which are as good or better than can be found in private industry and, on the whole, I would match our installations with any group.

Earlier in my talk I made a brief reference to those new ADP installations which are addressed to the basic mission of an agency such as flight and air traffic control weather predictions and the like, as distinguished from the administrative periphery in which we now find most of our applications. I hope we will see an expansion in this broader use of computers and also in the use of advanced scientific methods in the administrative area in order that the full potential of computers can be utilized.

Most of you probably remember the song that was popular a year or so ago which told the story of the ant that moved the rubber tree plant with little more than HIGH HOPES. Now I don't want to down-grade ambitious and optimistic thinking; in fact most computer salesmen follow such a philosophy and American industry thrives on it. In planning for a computer installation, however, we must combine our HIGH HOPES with some conservative thinking, including a recognition of the problems to be encountered and careful planning to avoid all of the known pitfalls. As public servants we have a responsibility to the taxpayers to see that the tremendous potentials of these new and expensive management tools are fully used and not abused.

ADP IN FEDERAL AGENCIES, II.

William A. Gill

William A. Gill is Chief of the ADP Staff in the Bureau of the Budget and serves also as Chairman of the Interagency Committee on Automatic Data Processing. His experience includes that of management analyst and consultant with the Federal government and private industry for more than twenty years. He has served as instructor or guest lecturer in several universities, has published several articles on management engineering subjects, and is co-author of a handbook on Systems and Procedures.

I should like at the outset to commend the Graduate School for conducting this seminar: first, because it helps to fill a need for educational opportunities of this character and, second, because I have heard of so many fine results from the earlier presentations of this seminar.

I am impressed by the fact that the word "automatic" is not used in the title of this seminar. It's called a "Seminar on Data Processing"-- not automatic data processing. This does not mean that you will not hear a lot about automation in data processing; there is ample treatment of the hardware itself; but it does mean that there is something to be gained by emphasizing the processing of data as a subject unto itself -- both with and without the hardware emphasis. This, I believe, is good. And I am going to grasp the opportunity right now to talk about the DP (data processing) without the A.

In this day and age, any presentation or discussion of the subject of data processing must be undertaken with a considerable element of risk. For one thing, "data processing" has different meanings to different people. For another, there are times when data processing is virtually worshipped as an end unto itself -- which is surely a fallacy. And thirdly, if we are not careful we may let ourselves become betwitched by the glamour of a relatively new piece of data processing hardware known as a computer, to the extent that we glorify this new device as something more than a tool for data processing. This also would be fallacious; indeed, it would be ridiculous.

Ordinarily, if there is mutual acceptance and understanding of what data processing is and what it is for, these risks either diminish or disappear. That being the case, it would be appropriate here and now for me to make clear to you what the term "data processing" means to me.

You most certainly are privileged to discard my definition and my concept of the purpose of data processing, but you need to know what they are if you are to grasp the true meaning of the things I am about to say.

The word "data" means facts; it means information. Data may be in the form of statistics, but do not have to be. Data can be received orally as well as in writing. Data can be obtained by visual observation of an object such as a building or a person as well as by the reading of a printed or typewritten page or the scanning of a chart, graph, blueprint, or photograph. As a matter of fact, we acquire much data by the feel of things when we want to test their weight, size, rigidity, texture, and temperature. Data, therefore, are factual information in any form concerning the past, present, and future, by whatever means of communication that may be used.

The word "processing" refers to the things we do with, or about, data. We request them; we generate them, manipulate them, consolidate them, integrate them, collate them, summarize them, and report them. We analyze them, evaluate them, extrapolate them. We praise them and damn them. Often we ignore them. And, to be completely honest, we sometimes hide them or disguise them. These are some of the things we do when we are processing data.

<u>Why</u> do we process data? If we were to explore this question in depth more frequently than we normally do, I believe we would have fewer data processing problems; we would make better use of the means we have to process data; and we would process less data, more useful data, and data that are more timely and more accurate.

So, before discussing some of the new tools of the data processing trade, some reference to the <u>reasons</u> for processing factual information seems to be in order, even though this means a restatement and review of certain truisms. Let's begin at the beginning.

An executive in any organization is faced with innumerable problems of many types at any given time. A large share of these problems will be peculiar to his particular organization -- problems, in other words, that are characteristic of the mission that his organization is seeking to accomplish. Very often, especially in Government, no other executive will be faced with the same set of problems. But a sizeable portion of the executive's problems may be found to be owing solely to the fact that he is an executive -- a boss, a director, an administrator -- a manager whose task it is to lead a group of subordinates, to make decisions, to make or to interpret policy; in short, to use his people to get results. In connection with common problems of this kind, one executive is no different than any other except that in dealing with problems that are common to all executives, some executives will be more effective than others.

These problems that are common to all executives at all levels of authority are plentiful. Many of them can be solved only on a temporary basis and with the guarantee that they will recur time and time again. Among these common problems are those that can be listed under such captions as Personnel Management, Financial Management, Supply Management, Employee Relations, Communications, and others just as familiar. And it is within the scope of one of these captions -- Communications -- that I shall aim most of my remarks today.

If we were to obtain a representative sample of all executives in an effort to find "Mr. Average Executive," I submit that we would prove the validity of two propositions I wish to make to you. The first I believe to be a fact. The second is an opinion or, perhaps more accurately, a theoretical conclusion.

My first proposition is that the average executive

1. Gets more data about his organization than he has time to use;

2. Gets considerable data that he does not know how to use;

3. Gets data that are dangerous to use;

4. Gets data that are useless;

5. Gets data too late to use; and

6. Fails to get much of the data he really needs.

My second proposition, admittedly theoretical but still an inescapable conclusion, is this:

An executive can be a good executive <u>only</u> if

1. He <u>KNOWS</u> what he (and his organization) have accomplished;

2. He <u>KNOWS</u> where he stands, in relation to where he planned to be at this point in time;

3. He <u>KNOWS</u>, based upon present status and past experience, the direction in which he should be moving and the momentum to use; and

4. He <u>KNOWS</u>, based upon past experience, that the quantitative and qualitative goals he has established for the future are logical and attainable.

In stating this conclusion, I use the word "knows" as distinct from such words as "believes, thinks, assumes, supposes, or imagines." The

connotation is that the necessary facts are in hand, in valid form, and in time.

Now, you have my two propositions. Assuming both are valid, when you consider them together you find that Mr. Average Executive has to be something less than a top-notch executive; at best, he is not as effective as he could be if there were communicated to him the data he must have to evaluate the past and the present on a knowledgeable basis and thus to make informed decisions concerning the present and the future.

This, I'm sure you will agree, is not a new situation about which people are just hearing. On the contrary, line and staff people at all levels of authority in Government and in industry have for many years struggled with the problem of keeping planners, policy makers, and decision makers adequately informed. But, as I have already indicated, the people we know as leaders have not on the average been informed adequately. Thus, they and the people who follow their lead have felt the unfavorable impact.

Why have we had to live with a situation such as this? A brief and simple answer to this question can be misleading, but let me make the attempt anyway. I believe there are three main contributing causes. First is the fact that, due in part to bigness, many of our organized groups contain so many layers or echelons of authority as to overtax the ingenuity of our data processing experts. Second, our efforts to educate the users and the producers of management data on how to handle those data have fallen short of the need, both in quantity and in quality. Third, the tools available for processing data, at least until the advent of the electronic computer, have been inadequate.

If my analysis of the basis for our data handling problems is substantially correct as far as it goes, then it appears that the problem of having an adequate set of tools for data processing is well on the way to being solved -- at least for a while. The first and second generations of electronic computers have been a means for coping with a host of data processing requirements in business management, science, engineering, and military operations -- requirements which in many instances could not possibly or feasibly have been met otherwise. So, while at this point we have no idea as to what succeeding generations of computers will look like or how they will perform, we can accept the fact that the data processing "break-through" that provided us with this entirely new and miraculous piece of computing machinery is an important milestone in history and for some years the computer will be a major tool of management.

But we still have the problem of education. Indeed, this problem is now magnified many-fold due to the electronic computer. When we take into consideration the "average executive" I described earlier and realize that it is now possible to reduce the amount of data which flows

to him by upwards of ninety percent; that the data he now might receive are in his hands in a few minutes or hours after the fact, rather than weeks or months; that in many cases the data will be substantially different in scope or format than previously; and that much of the data will come to him directly from the computer rather than through a succession of subordinates in the hierarchy -- these facts not only place a higher premium on training; they also seem to call for a new or different concept and emphasis in planning and conducting training programs of this character. I am sure that those who produce data through computer systems, those who use these data, and those who are faced with the tremendous training challenge these new systems pose, will jointly work out the ways and means for keeping training in step with the administrative and technological developments. As of now, however, it is believed that in the vast majority of organizations the use of the new tool has so out-distanced the training programs as to cause real alarm. If we are to use this new tool with wisdom, as we must, we have before us the urgent necessity for finding ways to close this serious gap. And if we truly meet the problem with adequate speed and objectivity, whatever we do must be something that recognizes that the gap that needs closing is not something we anticipate in the future. It is here now, it has been present in some organizations for years, it will take something more than the normal amount of aggressiveness to cope with it.

A few facts about our experience in using electronic computers in the Government may be helpful in two ways: (1) in identifying the underlying reasons why there are so many aspects of computer utilization that agency management needs to comprehend, and (2) in making it clear that the need for speed in orienting management is both valid and vital. As I recite these facts, however, three things must be borne in mind: First, meaningful experience in using computers is quite limited; you need the fingers of only one hand to count the number of Government agencies which have used computers more than five years. Second, we have had no experience at all with the so-called second generation of computers. And third, while some centralized information on the use of computers and the cost of their operation is now available, more is needed. With these limitations in mind, let me present some rather significant facts arising from our governmental experience. Within the time allotted I can do no more than identify each of twelve items. It is to be hoped that each of these items is considered in the development of training courses for executives and, during those courses, embellished and discussed to the extent that each deserves.

1. Where an agency has been job-conscious first, systems-conscious second, and computer-conscious third, our use of computers has so far been profitable. As you might expect, we find that in some cases the desire to acquire a computer has overshadowed other considerations.

2. The feasibility of acquiring a computer cannot be established casually or quickly. Unless adequate preliminary planning takes place and comprehensive and detailed systems studies are made, feasibility determinations will be based upon guesswork to a large extent, and precise equipment needs and specifications cannot be determined.

3. Although computers are ideally suited to statistical operations, financial accounting, and other administrative-type activities, particularly where mass paperwork is involved, this equipment is equally suited to other-than-administrative-type programs; and unless computers are considered in terms of all possible uses within an organization a true study of feasibility has not been made.

4. Almost invariably, the conversion of manual or conventional systems to computer systems requires substantial adjustments in the responsibilities and day-by-day work habits of key personnel, changes in organization structure (sometimes eliminating complete layers of organization), changes in both internal and external relationships, changes in operational policy and administrative regulations, and perhaps even in the statutes, all of which should be taken into consideration before the decision is made to acquire a computer.

5. The use of computers displaces people. So far, attrition has prevented the actual separation of employees through the process of reduction-in-force. This trend is apt to continue. With appropriate planning, and by keeping employees adequately informed, there seems to be no reason to expect that the morale of the work force will be adversely affected.

6. No department or independent agency in the Government is known to have reduced its annual requirements for money and manpower due directly to the use of computers. The reverse often is true. It is a fact, however, that in many organizations in the Government that use computers, substantial increases in workload have occurred without a corresponding increase in budgets.

7. Interagency cooperation in the use of computers has to date indicated that further exploration of possibilities in this area are most worthwhile. There are three main types of interagency cooperation. One involves the sharing of knowledge and experience and joint efforts to solve mutual problems. In this area the Interagency Committee on ADP has been quite active. A second type is the sharing of equipment time between agencies, with or without reimbursement. There appears to be considerable sharing of this type. The third type is the integration of systems between departments, when the systems of two or more departments bear a close relationship. One example of this is a case where one department approves a benefit payment, the Treasury Department issues the payment check, and the Post Office Department delivers the check to the payee. The possibilities for interagency integration of closely-related systems such as these are obvious. There are many similar possibilities that need exploring.

At this point it seems timely also to tell you that some months ago we in the Bureau of the Budget started a study to determine whether or not the need to establish one or more computer service centers in the Government truly exists. It has now been determined that there are sufficient requirements to keep such a center (or centers) busy. Therefore, as a further objective of this study we are proceeding with the help of appropriate agencies to draw up a set of definitive plans, specifications, and procedures as to how to organize, finance, staff, equip, and assign responsibility for the operation of an experimental service center. In connection with this planning we will deal also with such factors as service charges, economic feasibility, and competition with private business. We expect to finish this planning job in November of this year.

8. The vast majority of commercially available computer systems that have been acquired by the Government to date have been rented. The Bureau of the Budget and the General Accounting Office are jointly engaged in a project for the development of policy and guidelines on whether it is more advantageous to the Government, in all or some cases, to rent or to purchase ADP equipment. It is too early and too risky to forecast the outcome of this study, but I can and will say that there is a mounting body of evidence indicating that the practice of renting all ADP equipment as standard policy may need to be modified in some degree.

9. Compatibility of ADP equipment has been and is a sizeable problem; it is an expensive problem as well; but with appropriate concentration upon this problem by top governmental leadership, and with cooperation from the computer manufacturing industry, which seems assured, incompatibility should soon become one of the lesser problems in the ADP program.

10. From the inception of computers, the trend has been in the direction of large capacity equipment. Recently, however, there has come about a noticeable trend toward small capacity equipment. This had to come and there are many who will be helped by this new trend.

11. Computers, like all material things, are vulnerable to fire, flood, power failure, and to many forms of sabotage or attack by enemy agents or armed forces and their weapons. Since we are staking so much on the capabilities of our computers to process data which are essential to the continued operation of Government in wartime as in peacetime, we have a big job ahead to assure the preservation and continued operation of our equipment under all conditions and at all times. As in the compatibility problem, we have made some progress in dealing with the matter of vulnerability -- but we have a long way to go to solve this one. In this area we plan to launch a study soon.

12. One extremely vital question concerning the use of computers in the Government has not yet been answered as fully and as accurately

as it needs to be. That is the question as to whether or not the computer, as a tool, is fulfilling the expectations of those who advocate its use. In other words, is it practical in an operational sense, is it feasible in an economic sense, and is it desirable in terms of management's relationships with employees and with the public which the Government serves? We need a set of formulas and criteria for making assessments of this kind but at this point we have none that could be considered Government-wide in scope. Some of the agencies have begun to develop their own formulas and with the aid of their experience some Government-wide guidance can some day be formulated for assessing progress and results in the use of computers. Meanwhile, and as indicated earlier in my presentation, it seems quite proper to conclude that from an operational viewpoint the computer has established itself as a valuable and permanent tool of governmental management even though a few of the current computer applications are not the best examples we could find. In an economic sense, the computer "payoff" point is pretty difficult to identify. But on this point it must be remembered that with computers we can do many things that were desirable but not economically feasible to do before; we can do many things much faster and more accurately than before; we can handle increased volumes of work without corresponding increases in personnel; and, with all this, we can make our managers better informed than ever before. Can you put a price tag on these achievements? You can, of course, but whatever may be the price you determine, neither you nor your critics can prove or disprove it. In terms of its effect upon management's relationships with employees and the public, it is clearly evident that the computer has had and will continue to have a distinct impact upon these relationships. Our experience to date with this kind of impact has on the whole been good. But through that experience we have learned that we have a long way to go to insure that these relationships remain good.

With the limited and informal evaluation measures we have been using so far, some additional facts have been developed that need to be passed on to groups like yours. These are facts that serve to point up certain pitfalls which, whether or not already known to you, you will want to avoid. More specifically, I refer to things like these:

1. Almost invariably, ADP costs are underestimated. Variances run as high as 200 percent over budget expectations. This is due to several causes such as underestimation of machine time required; failure to allow for set-up time, down time, and maintenance; and the fallacious assumption that workload will remain constant, which rarely occurs.

2. The purposes behind the acquisition of a computer are often so vague or general as to leave the impression that the real purpose is to get the computer first, then figure out how to use it or what to use it for. In enough cases to cause alarm, this is substantially what has happened. I am sure you will agree that before we get computers we ought to have specific plans for their use.

3. Computer programming, systems analysis, and the writing of instructions, with rare exceptions, take longer than expected -- sometimes over twice as long.

4. There seems to be a general tendency to overlook certain cost items when estimates are made; for example, supporting equipment, building modification, training, and travel.

5. Punched-card equipment due to be given up when the computer is installed somehow is not released in some cases. And in many cases punched-card or manual systems due to be discontinued are not, so we end up with duplicate systems.

6. In cases involving parallel operations, that is, when the new ADP system and the old system operate concurrently until the new system is thoroughly de-bugged, we find that more often than not the period of parallel operations is excessive -- six months, for example, when one or two months would be ample.

****

Please don't get the idea that our findings are all as unfavorable as the ones I have been describing. On the contrary, there are numerous ADP applications to which we can point with some measure of pride. Many novices and neophytes in the ADP field are consistently being urged to visit these and other "going" operations in order to see at first hand how computers have been geared up to do a creditable job.

Certainly, not all ADP applications will always be models of efficiency. But we are learning fast and we firmly believe we have occasion even now to be confident that on the whole, throughout the Federal Government, we are going to do even better than a creditable job in applying ADP equipment to our operations. I feel certain that you as individuals and the departments you work for will make significant contributions to that ambitious goal.

All of what I have so far said can be added up to mean that ADP is a very important set of initials for executives in Government and elsewhere to consider. <u>BUT</u>, as we go about giving consideration to these initials, let's change the order. Let's put the DP first. Let's establish the essentiality of the data we believe are needed; let's establish who needs them, when they need them, why they need them, and the form in which they can best use them -- that is being job-conscious. Next, we're ready to be systems-conscious; to establish <u>how</u> we shall produce the data required. Finally comes consideration of the equipment which is needed and is best suited to automate data processing systems.

There is, of course, little if any difference in the generic meaning of the terms "Automatic Data Processing" and "Data Processing-Automated."

Whichever term we may use, the most important consideration is the psychology and methodology of our approach. And if the initials are indicative of approach, the moral of my story is: let's first consider the job in terms of its essentiality; next, the system in terms of its efficiency; and last, the equipment best suited to meet the needs of the system.

There just isn't any other more logical approach.

PART 2

INFORMATION SYSTEMS EQUIPMENT

THE ADP FAMILY AND THEIR GENERAL CHARACTERISTICS

M. H. Hansen and J. L. McPherson

Morris H. Hansen is Assistant Director for Research and Development of the Bureau of the Census where he has been concerned primarily with research, maintenance of high statistical standards, sampling, quality control, and applications of electronic equipment. He is a Fellow of the American Statistical Association, the Institute of Mathemtical Statistics, and the American Association for the Admancement of Science, author of numerous technical papers, and joint author of a text on sampling methods and theory.

James L. McPherson is Machine Development Officer at the Bureau of the Census, with previous experience with the Works Progress Administration and the U.S. Public Health Service. Since the mid-forties he has fostered the application of electronic data processing equipment to Census problems, played a major role in the acquisition of the Univac System by the Census, and has been concerned most recently with the development and application to Census problems of FOSDIC.

At the 15th International Congress on Hygiene and Demography, in a paper relating to tabulation facilities at the Bureau of the Census, Mr. H. T. Newcomb said:

> "Too much cannot be said in praise of this machine which has enabled us to compute results with much greater rapidity and accuracy than by the old method of tallying besides giving the opportunity to make a much more thorough analysis of the figures."

Mr. Newcomb did not refer to ADP, probably because he was not familiar with that particular verbal short hand. This unfamiliarity is understandable. The meeting at which Mr. Newcomb made this remark was held in September 1912, over 48 years ago.

Mr. Newcomb was talking about the then relatively new punched card tabulating equipment. He and his coworkers were undoubtedly just as impressed with and optimistic about the potential contribution of punched card tabulation equipment as we, their descendants at the Bureau of the Census, are about electronic data processing equipment. Certainly if it had occurred to them these forebears of ours would have been entitled to call their new devices ADP machines. The equipment invented

by Herman Hollerith during the 1880's constituted just as much a break-through in his day as the electronic equipment which is the subject of these seminars today.

Many of us now on the staff at the Bureau of the Census can be likened to members of the D. A. R. As we understand them the members of this organization take no small amount of pride in the fact that one or more of their progenitors participated at the birth of our great country. Admittedly, however, none of the present members threw tea in Boston Harbor or pulled an oar at Valley Forge. Similarly none of us at Census punched even a single hole in an 1890 Census punched card or attached a wire to the original Hollerith tabulator. Nevertheless we are proud that Dr. Hollerith was a Census employee during the processing of the 1880 Census where he recognized a pressing need for more automatic data processing equipment than then existed. Like the D. A. R. we enjoy wrapping ourselves in the glory of our predecessors.

Certainly history has justified the enthusiasm typified by Mr. Newton's comment. A giant industry has grown from the invention of punched card equipment. Although we wouldn't know how to go about proving it we believe we can even today safely affirm that throughout the world, more data processing activity is accomplished with punched card equipment than with any other kind of machinery, including high speed electronic data processing equipment.

One thesis we want to develop here, for which the foregoing is a brief background, is that ADP is neither new nor has it really been achieved if one interprets the A, for "automatic," at all literally. Data processing means many things to many people. The most nearly completely "automatic" data processors with which we are familiar are devices which in mid-20th century terminology are called analog computers. The thermostat on the wall in your living room is a good example of such a device. The data it processes is the temperature. The purpose to be served is the maintenance of a relatively constant temperature. Except for infrequent malfunctions, which can usually be corrected readily, this data processor is truly automatic in the sense that it requires effectively no human attention after the initial parameters of the problem have been stated -- i.e. the desired temperature has been communicated to the data processor by setting a dial.

It is, however, the digital or general purpose type of data processing equipment that is the subject of this seminar rather than specialized analog computing equipment. Certainly modern electronic data processing machines make it possible to "automize" some aspects of data processing to a much greater extent than was possible a decade or so ago. Nevertheless there continues to be a significant amount of work requiring the attention and skills of human beings involved in the successful use of these equipments. From this point of view then there remain many

challenging areas in which research and development may bring us still closer and closer to full "automatic" data processing.

We mentioned the invention of the punched card method toward the end of the last century as one bit of evidence that the development of machinery to process data is not exactly new. Most good reference books on modern electronic computers pay homage to Charles Babbage. As early as 1812 Babbage began work on a device he called a "difference engine," and from 1833 until his death in 1871 he devoted himself to the design and construction of an "analytical engine." Today competent engineers assure us that Babbage's design was a good one and incorporated much of the logic we find in modern electronic computers. Babbage was unsuccessful because in his day parts could not be machined to the tolerances he specified, not because of any defects in his design which incorporated the three main elements which characterize today's computers -- namely an arithmetic unit, a control unit, and a memory. Incidentally -- and we wouldn't be surprised to learn that Hollerith was aware of this -- Babbage proposed to use punched cards for input, particularly to the control unit, in his analytical engine.

Thus, we submit, the work of Babbage and of Hollerith, as well as the work of others who developed desk adding machines and desk calculators, not to mention the oriental who a thousand or more years ago invented the abacus, is evidence that the search for devices to "make automatic" in some sense or other the processing of data is not new. What is, in our opinion, new is the machinery that has grown from the marriage of the mathematician, statistician, accountant, and others with data processing problems to the electronic engineers who were capable of organizing the products of their industry into the complicated networks of circuits which make the electronic data processors of today.

Just who deserves credit for first recognizing that the vacuum tubes used in radio transmission and reception had properties which would make them useful devices for computation we are not qualified to say. Dr. Norbert Weiner, of Cybernetics fame, pointed out that vacuum tubes could be used to perform binary arithmetic prior to 1940. To the best of our knowledge the first men actually to assemble equipment electronically to compute were Dr. John Mauchly and J. Presper Eckert at the Moore School of Electrical Engineering at the University of Pennsylvania. During the early 1940's these men developed and built the Eniac (Electronics Numerical Integrator and Calculator). It was built for the Ballistics Research Laboratory at Aberdeen Proving Ground and was completed in 1946. The Eniac, although it has general computing ability, was a rather special purpose computer. Its main function was to compute firing tables. For this purpose it was eminently successful. It computed a trajectory in about half the time it takes a shell to travel from the gun to the target.

While they were still building the Eniac during World War II, Eckert and Mauchly recognized the potential value of electronic computers for general data processing purposes. They began to formulate, at least in their minds, the design of a general purpose electronic computer. Dr. Mauchly's main interests were in astronomy and physics. Mr. Eckert was an electronic engineer. Neither of them was particularly familiar with the problems of accountants, economists, social statisticians, and others with large data handling responsibilities not in the domains of physics and engineering. They sought the advice and consultation of persons with experience in mass data handling and among other organizations they visited during the early 1940's was our Bureau of the Census.

Their early visits were frankly for the purpose of finding out the kinds of problems we faced and understanding what machinery we were using to solve them. It was generally understood that their endeavors in the area of electronic computing machinery would have to be devoted to equipment to facilitate our war effort and that not until hostilities ended would they be able to devote any important share of their time to the development of equipment for civilian use. During this period our attitude at the Census was a peculiar combination of awe and doubt. On the one hand it was obvious that these gentlemen were respectable citizens, presumably responsible, since they had a contract through a major university to build equipment for the United States Army. On the other hand they seemed to have the crazy notion that somehow we could use our nine tube superhetrodyne radio receiver to tabulate census results. We willingly answered their questions about our work and showed them our punched card processing equipment. We now fear, however, that in those days we classified them as just a couple of sightseers.

When the war ended, however, Eckert and Mauchly made serious representations to us about building electronic data processing machinery for our use. It then became important for us to decide just how seriously their proposals should be taken. There existed in the mid 1940's a "Science Committee" composed of representatives of the various bureaus which make up our Department of Commerce. At our request this committee met to hear us describe the proposals we had received from Eckert and Mauchly and to advise us on a way to proceed to evaluate the validity of their ideas and their competence to implement them. It was the recommendation of the committee that scientists at our sister agency in the department, the National Bureau of Standards, could be of real help in understanding and evaluating the technical aspects of the proposed electronic computing equipment. Dr. Archibald McPherson, Deputy Director of NBS, was a member of the Science Committee and played a major role in initiating a cooperative arrangement between our Bureau of the Census and the NBS which continues to this day.

After preliminary meetings at which as best we could, we familiarized the mathematicians and engineers at NBS with the general character of our work, it was arranged for Eckert and Mauchly to describe their proposed equipment to a group of NBS scientists. We were pleased and relieved when

our colleagues at NBS told us that we were not being duped by a couple of charlatans who wanted to sell us the Emperor's new clothes but that electronic computing in the Eckert and Mauchly way was indeed a real possibility and that these two inventors had made a very good impression at NBS.

In 1946 we transferred $300,000 of our funds at Census to NBS to defray the cost of work they were doing in our behalf. Here as an aside we might report how the fates and the Congress of the United States were good to us. As you know our war effort had moved large numbers of people in our country; many people had migrated from one area to another to make their contribution at defense plants. It was decided that a population sample census to be enumerated in 1946 should be taken and Congress had earmarked in our fiscal 1946 appropriation funds to defray the costs of getting ready for this Census. When it came time to pass our fiscal 1947 budget, however, Congress changed its mind about the desirability of a population sample census and did not appropriate funds to conduct it. Fortunately, however, they did not impound the get-ready money they had supplied the previous year and we had no difficulty in getting their permission to transfer $300,000 to NBS.

Some of this money was used by NBS to pay for a study contract they awarded to Eckert and Mauchly in 1946. By this time these men had left the University of Pennsylvania and established their own organization. One result of this study contract was a general design of the computer now known as Univac 1.

While the study contract was in being other organizations, particularly the Air Force, the Army Map Service, and the Office of Naval Research, developed interest in the possibility of electronic computation and sought the advice and assistance of the NBS personnel with whom we were working. These defense groups were able to finance work on a much larger scale at NBS than we were able to underwrite. A large group of competent scientists were assigned responsibility at NBS in this new and interesting field.

In June of 1948 the National Bureau of Standards entered into a contract with the Eckert-Mauchly Co. for the construction of a Univac for Census. Shortly thereafter the contract was supplemented with an order for two additional Univac systems, one for the Air Comptroller and one for the Army Map Service.

The original delivery date specified was February 1, 1950. The first Univac ever built was actually accepted by the government on March 31, 1951. Incidentally it is still in use today at our Bureau where it is performing quite satisfactorily. In 1954 we acquired a second Univac 1 and last year we added two Univac 1105 systems at our Washington Offices and made cooperative arrangements for the installation of compatible 1105 systems at the University of North Carolina and at the Armour Research Foundation of the Illinois Institute of Technology. At each of

these institutions we will have first priority on from one-half to three-fourths of the computer time during the next two years which span the period of our peak requirements for processing the 1960 Decennial Censuses of Population and Housing.

The period from 1948 to 1951, during which the government had contracts for general purpose digital computers but no hardware supplied by private enterprise, was certainly stimulating and some times trying. The Congress maintained a lively interest and followed the development closely. Great pressure was exerted on NBS personnel by our Bureau of the Census, by the Air Comptroller, by the Army Map Service, and by the Congress to "speed up" the production of the equipment, although in retrospect we all now recognize that just how they should go about effecting a speed up was an administrative detail we were quite willing to leave up to them. They on their part worked with a will and although they didn't find any magic formula to make contractors deliver on schedule they accomplished a near miracle by designing and building SEAC. This computer actually began producing useful computations on May 9, 1950 almost a year before the first Univac was accepted. This was a real accomplishment, particularly in view of the fact that this endeavor began at NBS after the Univac contracts had been awarded.

During the early 1950's many organizations entered the electronic computer field. Most notable was IBM. The energy and enthusiasm with which they worked in this area once they had decided to enter it was truly admirable.

The first machine in the 700 series, the 701, was delivered in 1953. Thereafter its successors came along in rapid succession: the 702 and 704 in 1954, the 705 in 1955, the 709 in 1958, and just in the past year they are offering the solid state 7070 and 7090.

SOURCE DATA AUTOMATION

Robert S. Malone

Robert S. Malone is in charge of the Navy Source Data Automation Program which he developed and which is one of the functions of the Data Processing Systems Division. He has served the Navy Management Office for approximately 11 years, having prior experience in forms management, correspondence management, and work simplification for the Veterans Administration and 8 years with the National Park Service in administrative and management work.

What is Source Data Automation?

With all the discussion about electronic data processing systems, we too often overlook one of the major problems confronting the development of really effective information systems -- the automation of the basic paperwork operations which are the foundation of these systems.

Source Data Automation (SDA) simply means applying the techniques of automation to the source areas -- where information begins. If we are to reap the full benefits of automation, it should be applied to the whole range of an information processing system -- not to just the end product areas. It can be as important for management to have routine paperwork operations mechanized as it is to bring a high degree of automation to the large data processing shops where information is ultimately collected and summarized.

How is this done? By recording data -- the first time they occur -- into a machine "common-language" as the automatic by-product of the initial operation. This "common-language" may be punched paper tape, edge-punched cards, EAM punched cards, or punched tags. Data so recorded then become self-perpetuating, and they may be used over and over to satisfy the many and varied requirements that exist in almost every system.

By using office type equipment -- such as typewriters and adding, bookkeeping, calculating, and addressing machines -- which have been adapted to use "common-language" media -- the basic paperwork necessary to every system can be prepared with a minimum of manual operation. By using tape-to-card and card-to-tape converters, communication equipment, and electric and electronic data processing equipment, the automatic handling of data can be continued on throughout the system, whatever the requirements.

Thus, by using the proper combination of machines which can read and write a "common-language," SDA can be tailored to fit most any systems requirements -- from the automatic typing of a form to the complete mechanization of a system.

SDA provides, to a large degree, the means for processing data from machine to machine, rather than from person to person. And with its building block approach, it provides a high degree of flexibility -- it can be modified as requirements change. SDA not only integrates basically dissimilar machines into a coordinated machanized system, but more importantly, it is a method for giving managers complete and timely information on all aspects of the function for which they are responsible.

Where to Use Source Data Automation

The techniques of SDA can be of great benefit to those systems where one or more of the following conditions exist:

Repetition -- in many systems a large part of the data is constant; it is continually reproduced from day to day and from document to document. Examples where much of the data are constant are the reordering of stock supply items and maintenance parts and processing the same names, addresses, and descriptions.

Volume -- In many systems the sheer size of the operation makes it almost impossible to handle the volume efficiently when only manual methods are employed.

Deadlines -- Speed in processing data is often of the greatest importance. Data should be received while they are current. This frequently cannot be accomplished using conventional manual methods.

Errors -- Errors in the processing of data are often one of the most serious problems. In manual recording and processing of the same data at various points in a system, errors are difficult to control. Processing of data in machine language can be made error-proof.

Bottlenecks -- Very frequently there are operations in systems that can't keep pace with other related operations. Appropriate automation may break these bottlenecks.

How to Determine Applicability

In many respects the analysis of a system to determine SDA applicability is no different than other systems studies. However, certain considerations frequently take on added significance insofar as SDA is concerned. Some of these are:

A complete survey of the entire system is a must. An SDA application generally takes the processing of data across divisional lines. This makes it imperative that the analyst fully understand the total system. Furthermore, a complete study will serve as a catalyst to complete understanding and support of SDA by all those personnel to be affected.

A detailed time and cost analysis of the present and proposed system is usually essential. Some money will have to be spent on equipment to make an SDA application. Even though the economic aspect is not always the most important factor, very frequently a system must be justified on that basis.

A complete analysis of all the details in a system must be made. Flow charts showing every step in the system are very helpful. Since machines are being asked to talk to each other, the machine language prepared at one point must be exactly as required at every other point of use. The omission of any details can cause an application to fail.

The need for by-product information at any of the basic processing steps in the system should always be considered. SDA readily permits the collection of by-product information in machine language for further automatic processing. This is an area that frequently results in the biggest savings in an SDA system and should always be carefully checked.

Know as much about the equipment capabilities and limitations as is practical. This doesn't mean that it is necessary to be an equipment expert, but it is essential to have a good working knowledge of the various pieces of equipment.

Take into consideration the exact location of the equipment. Some SDA equipment is quite noisy and should be placed only at certain positions within an office. Space rearrangement may be necessary so that equipment can be placed where it will not interfere with the office personnel.

Check out the type of data processing equipment already in use at your activity to be sure that any new equipment will be compatible with existing equipment. Can the machine language produced by a piece of SDA equipment be used in an already existing central data processing installation?

The training of personnel to work in a SDA system should be considered. A brief amount of training will be necessary for personnel who will operate the equipment. Depending on the complexity of the system and the equipment used, some training in programming will be necessary. However, usually no more than five days of instruction is required. Most equipment manufacturers provide this training.

An SDA system should be tested before installing it on a broad scale. A good "shake down" or debugging can be very important. Try to test a system before buying equipment. This can be done by leasing equipment. This will give you a good amount of time to check out the system before spending the money for purchase. In some cases, short periods of debugging can be accomplished on the manufacturer's equipment.

There is more need for analyst assistance in the installation of an SDA system than in the introduction of many other systems. You cannot merely submit a report and expect the operating offices to install. SDA is too specialized for the operating offices to install and to get into proper operation without assistance from the analyst. Follow-up and review of the system after it has been in operation are also important.

The question of organization must also be considered. SDA does not generally require any major organization changes. However, in order to get good utilization from the equipment, centralization of functions may be required.

The proper design and construction of forms is a very important consideration in a Source Data Automation system. Efficient forms design involves (1) a detailed analysis of the system, and (2) a knowledge of the features of the equipment that will be used to automate the system.

Type of Machines

The following are several general characteristics of SDA machines:

1. They are adaptions and modifications of the basic office machines we have ssen for years, e.g., electric typewriters, adding machines, accounting machines, and calculators.

2. What they do and how they do it are relatively easy to understand; consequently they are easy to operate and require little operator training.

3. They are relatively low in cost.

In addition to the general characteristics of SDA machines, here is basically what SDA machines do:

They allow information entry through keyboard (input).
They read a machine language (input).
They usually print on hardcopy (output).
They produce a machine language (output).

When accomplishing the above actions, SDA machines perform specific functions with data, e.g., recording, adding, reading, calculating, writing, converting, and transmitting. Therefore, we divide the machines used in SDA into various groups in order better to understand the functions that the machines perform. However, these classifications are not so rigid that a machine which is found in one group cannot perform a function similar to machines found in another group.

Machine Language

The objectives of SDA are to reduce repetition and error, increase speed, and provide better information by processing data from machine to machine automatically. To attain these objectives, coded machine languages are essential.

Punched Cards -- Currently, the two most popular machine languages for SDA are the punched card and punched paper tape. The punched card has been used for many years by Navy in central data processing shops and has provided us with a means for accurate and timely reporting. In the future, the punched card will be used very extensively in source data applications.

Punched Paper Tape -- The use of punched paper tape as a machine language for SDA has been popular since 1954. Prior to this date punched paper tape applications were largely confined to the communications industry. Today, a wide variety of office equipment accepts and produces punched paper tape as a machine language.

The most popular paper tape language in office automation today is the 8 channel tape. It gives the many code possibilities needed for the alphabet and numerals and the many machine functions required for modern automation.

There is a striking similarity between the 8-channel tape code and the 80 column card code. This relationship is not accidental. It provides us with a means to convert tape-to-cards and cards-to-tape with a minimum of difficulty. Therefore, conversion provides us with flexibility in choosing the machine language to meet our requirements.

Edge Punched Cards -- To provide ease in handling, filing, identifying, etc., tape codes can be punched into cards of various sizes and shapes. This technique is referred to as edge punched cards or "wide tape."

Other Languages -- Recent developments in machine languages include print-punch tickets, magnetic tape, magnetic ink character recognition (MICR), optical code scanning, and optical character recognition.

Some of the Advantages of SDA

Increases speed in processing information; provides greater accuracy of information; saves personnel costs and operating time by reducing typing time, proofreading, and other processing time. Makes more efficient systems and provides better information.

SDA has a wide variety of applications, and it often offers significant benefits to managers who might never hope to justify a costly electronic data processing system.

Systems

Following are some of the systems to which SDA has been successfully applied in Navy:

- Officer Appointment Documents
- Commercial Procurement
- Transportation Procedures
- Military Construction Status Reporting
- Civilian Marine Personnel Paperwork
- Self-Service Supply Operations
- Production and Time Recording
- Ammunition Stock Recording
- Recruitment of Enlisted Personnel
- Machine Shop Loading System

DATA COMMUNICATIONS

L. L. Griffin

Lindon L. Griffin is an electronic engineer in the Data Processing Systems Division of the Navy Management Office. This office provides staff assistance to the Secretary of the Navy in the development and management of data processing and data transmission. Mr. Griffin has been engaged in commercial and military communications since 1929. He is registered as a professional engineer of the District of Columbia and is a member of the Institute of Radio Engineers.

Introduction

In earlier sessions of this series you have been told something of the features of Electronic Data Processing Equipment. The previous speaker has discussed some of the methods and equipment used to record information at or near its source and in a form that can be manipulated, processed, and re-used with a minimum of manual effort.

In many government and commercial organizations it is often necessary to communicate information recorded at one location to another location for one purpose or another. An example is a headquarters or control office with a number of geographically dispersed field offices in which information flows primarily between the control office and the various field offices. I'm sure you can think of many other examples.

The traditional method of carrying on such communications has been by mail, or, when the information was needed in a hurry, by teletypewriter transmission or Western Union message. No particular problems of compatibility arose because the information documents were usually in written form and processing was essentially manual in nature. If electrical transmission was used, manual processing was necessary either to accomplish actual transmission or to convert the information into a form suitable for transmission by automatic means. Punched paper tape is of course the outstanding example of automatic transmission methods. The characters, functions, and symbols employed in teletypewriter transmission have been pretty well standardized for quite some time. Most of the teletypewriter compatibility problems which have arisen involved machine interconnection capability rather than character coding or recording. Some interconnection problems, however, still exist, particularly in international operations.

When an organization embarks on a program to produce source records in a machine language form and to process these records in an Automatic Data Processing system, compatibility of character codes and recording media between the various machines involved must be considered. If electrical transmission is to be used for communicating records, three-way compatibility between source records, communication system, and ADP must be considered.

Compatibility Problems

You probably already know that the growth of automatic data processing has sparked the development of a variety of data recording methods. Punched paper tape, punched cards, and magnetic tape are well known methods. Automatic reading of printed characters is under intensive development with some systems already in use. Magnetic card and photographic film record reading are on the horizon. Paper tape is available in more than one width and the number of tracks or channels varies from five to eight. The teletypewriter code is almost always used with 5 track tape. There is, however, no generally accepted standard code for 6, 7, or 8 track tape. The work "code" as used here means the sequence of zeros and ones which identifies the various characters, symbols, and functions used for information recording.

The same situation exists in magnetic tape recording. It is unusual to find two manufacturers using the same code. In some cases, different systems produced by the same manufacturer do not use the same codes. Magnetic tape widths of 1/2, 3/4, and 1 inch are used. Track arrangement, number of characters recorded per inch, and a number of other factors can affect the compatibility of an ADP system.

Two incompatible punched card systems are also in general use today.

I have said enough, I'm sure, to indicate that compatibility problems can arise in data communications between the machine record sources and the processing centers.

Compatibility through Integrated Systems

Equipment manufacturers try to solve the compatibility problem by offering what some of their salesmen call an "integrated" system. What they mean is that they will provide not only the EDP equipment but the source data recording equipment and the communication terminal equipment as well if electrical communications are to be used in the system.

The IBM corporation, for example, is basically a punched card oriented organization. A few years ago they developed a transmission terminal device called the IBM Data Transceiver for use in communication service between processing centers and source data points where IBM punched card records are used. More recently they have added magnetic

tape transmission equipment for use with IBM magnetic tape recording and processing equipment. Arrangements can also be made to connect paper tape reading and recording devices to some versions of the communications equipment for transmission of suitably encoded punched paper tape.

The Radio Corporation of America (RCA) is also marketing punched paper tape transmission equipment called DASPAN. The tape read and produced is, of course, compatible with RCA's data processing and source recording equipment.

Other manufacturers either offer their own communications equipment or are prepared to make suggestions on equipment they consider suitable for a customer's needs.

Hence one way to achieve compatibility is to procure all of your equipment or systems from a single manufacturer and let his representative solve the compatibility problem.

Compatibility through Standards

Large users of EDP do not want to be restricted to the use of a single manufacturer's equipment in order to achieve data recording, communication, and processing compatibility. Neither do they like to pay for additional equipment and operator time to convert data from one form to another in order to intercommunicate between unlike manufacturers' equipment. Consequently there is considerable emphasis at present in industry and government on the adoption of standards for tape recording governing such factors as physical dimensions of tape, coding of characters, and record track arrangements. Because of the immediate and pressing need, the Department of Defense has moved ahead with the adoption of a standard for use in data communications and in data input/output devices. This standard is the "Fieldata" code. It has been approved by each of the services and is now being printed for distribution. Several industry association committees are also actively considering various codes, including the Fieldata code, with the objective of establishing industry standards. Considerable research has gone into the development of the Fieldata code. We are, of course, hopeful that this code will provide the basis for a governmentwide and industry standard. This standards problem has also been the subject of discussion in a group concerned with the establishment of international standards.

Communications Methods

My remarks today are directed primarily at electrical communications in EDP. This is not intended to imply that you must have electrical communications because of EDP. Many activities employing EDP equipment do very well without extensive use of transmission facilities.

A spokesman for a major southern railroad, speaking at a seminar two years ago, described their use of rail facilities to obtain overnight delivery of machine records from all parts of the rail system to their central data processing center. Regular mail, air mail, and air express are all being used successfully for data record transfers. Electrical transmission of data records will cost more, in general, than will mail or other transportation service. Therefore, presumably, you will use electrical transmission only when the urgency of need or other factors justify the extra cost.

Assuming you do have data to transmit, you should of course select transmission terminal equipment capable of accepting and reproducing data in the proper form for your needs. If your source data are in perforated paper tape form, you will no doubt want punched tape transmission equipment. If your data records are 80 column punched cards, you may want punched card transmission equipment. It may sometimes be advantageous, however, to employ card-to-tape conversion equipment at source data points in order to make use of existing paper tape transmission facilities. I know of a number of such applications in government agencies and industry.

You may find that you have a choice of transmission devices designed for the same type of transmission. They may differ in such factors as transmission rate and number of automatic features available.

Transmission Rates

In general the more automatic features an equipment has, the more it will cost. Faster transmission rates also mean higher equipment cost in most cases. Paper tape transmission equipment can be obtained to operate at rates of six to one hundred characters per second. The top limitation is the speed at which the mechanical paper punch can be operated. Teletypewriters which read and reproduce paper tape at six to ten characters per second can be rented for $60 per month or less. Sixty to one hundred character per second equipment is becomming available with an indicated price range of $400 to $600 per month.

An IBM punched card transceiver to operate at a rate of 11 eighty column punched cards per minute rents for about $200 per month for 8 hours use per day. Collins Radio Corporation offers an equipment designed to transmit 100 punched cards per minute in both directions simultaneously over a two way line. Rental cost of this equipment will be approximately $2,000 per month. I should add that the Collins' transmission equipment makes use of IBM card reader-punches as input/output equipment.

Transmission of digital information directly from magnetic tape in business type data processing is relatively new. Both Collins and IBM have announced equipment for this purpose. There will be other

announcements. Transmission rates between magnetic tape terminals are usually limited by the interconnecting circuits' transmission capabilities. Rates on the order of 200 characters per second are being tested on leased telephone lines. Costs of equipment are not low but may eventually be reduced. The cost of a magnetic tape transmission terminal at present is on the order of $1,500 to $2,000 per month.

There is considerable discussion in industry at present on the subject of direct computer-to-computer transmission over communication circuits. A number of manufacturers are known to be working on terminal devices to accomplish such transmissions, particularly in connection with small scale EDP systems.

Transmission Channels

Telegraph

The commercial communications companies are, of course, in the business of providing transmission circuits. In the Continental United States this reduces to the Telephone Companies and the Western Union Telegraph Company. The usual types of circuits, for business data transmission, are of telegraph and telephone grade. The basic difference between the two, from an operational standpoint, is the transmission rate that can be used. Early telegraph was, as you probably know, manual Morse transmitted by a hand key and copied by ear. Morse telegraph has largely been superseded by the automatic telegraph or what is commonly called the teletypewriter. Early models of teletypewriters operated at about 40 words per minute or 4 characters per second; modern machines at a maximum of 10 characters per second. Manual Morse speeds were lower.

Such low transmission rates do not use the transmission capability of a telephone circuit efficiently. Consequently, over the years, the communication companies have found ways to sub-divide a telephone circuit into sixteen or more low speed telegraph channels. The American Telephone and Telegraph Company's basic rate for a leased telegraph channel for 24 hour per day use is $1.45 per mile per month. This rate applies for 60 word per minute (6 characters per second) half-duplex service. Half duplex is telephone jargon meaning send or receive capability but not simultaneously. Ten characters per second (100 words per minute) transmission and simultaneous send-receive options cost more. Ten characters per second is the maximum rate used on telegraph channels. The per-mile rate drops somewhat for distances over 250 miles. This rate is for interstate long line service. Operating company rates for such service within their operating area may vary somewhat from this figure.

You can also subscribe to the AT&T Company's Teletypewriter Exchange Service (TWX) if you wish. The teletypewriter with tape read-punch capability rents for somewhat less than $100 per month. You then pay for the number of calls you make to other subscribers on a time and

distance basis much as for long distance telephone calls. With TWX you are limited to transmitting the characters, symbols, and functions prescribed for such service. You would not necessarily face the same limitations with leased private line service.

Telephone

The use of telephone grade channels for business type data transmission is growing rapidly. Considerably higher transmission rates can be used on telephone channels than is the case on telegraph channels. Sixty to one hundred characters per second are becoming available in paper tape equipment. IBM magnetic tape equipment will operate at 150 characters per second. With somewhat complex signaling equipment, a transmission rate of 200 characters per second or higher may be possible. Most of the digital data transmission equipment designed for use on telephone channels have either been recently announced or are still in the development stage. The exception and most widely used equipment is the IBM punched card transceiver which has been in use on leased telephone lines for four or five years. In March of this year the Company announced that the transceiver is now available for use on telephone lines on an intermittent or "message toll" basis. Message toll is a new type of service that seems destined to be extremely popular. Essentially message toll consists of placing a long distance call by telephone and substituting data transmission terminal equipment in lieu of the telephone subset when the connection is completed. Charges are based on time and distance as with TWX and telephone calls. Message toll type service has not yet been established in all areas of the Continental United States according to the latest information available to me.

Charges for a leased telephone channel begin at $3.00 per mile per month under AT&T long lines rates. The rate is lower for distances in excess of 250 miles between terminals, dropping by successive steps to $1.00 per mile for all circuit miles in excess of 1,500.

It may be cheaper to use telephone channels in lieu of telegraph channels for data transmission where relatively large volumes of data are involved since the rate at which information can be transmitted on telephone channels can be 10 or more times that of telegraph channels.

Over-all Transmission Costs

It may seem trite to say that lowest over-all cost should be the primary objective in transmission planning. Lowest over-all cost may be very difficult to determine. Selecting relatively sophisticated high-speed terminal equipment, for instance, may reduce line use time and hence keep line costs down. The sophisticated equipment, however, will cost more than simple slower equipment. The use of transmission codes which facilitate error or garble-detection and automatic features to control errors will increase terminal equipment costs but may save many

dollars through over-all speed-up of data processing operations and through a reduced need for retransmission of erroneous data. Time permits only such suggestions as to the difficulties of obtaining an optimum solution.

Transmission Error Control

Control of errors during transmission is an important consideration when information is to be transmitted in data form. Occasional minor-garbling, such as a wrong letter in conventional narrative type messages, usually does not present a serious problem. The redundancy inherrent in the English language permits considerable leeway for reconstruction of the sender's intent or meaning. A practice long used with narrative message transmission is that of repeating key parts of messages such as numbers either individually as they appear in the text or later at the end of a message in what is called a confirmation line. This is an example of an error control measure which has been in use for many years. It was up to the reader of the message to detect errors by matching the first and second transmissions. Transmission of information in data form presents some new problems or perhaps the same problems in a greater degree. First, the redundancy present in word messages is not present in data; and secondly, the reader is a machine possessing, we hope, a lower degree of intelligence than human beings. Hence, we either need transmission systems that are relatively free of garbling or we need automatic means of detecting when garbling has occurred.

Transmission systems, even the internal ones in data processing equipment, are not completely free of garbling. As a result, most data transmission systems and, I might add, most data processing equipment, employ a means of detecting when garbling of characters occurs. The method most frequently used is to add to each character a controlled amount of redundancy. This redundancy is usually in the form of a "parity bit." It is injected automatically by the recording or transmitting equipment as necessary to make the "ones" bits in each character total an odd number such as 3 or 5 if an "odd parity" system is used. In an "even parity" system, it would be added to make the total, 2, 4, 6, etc. The "parity bit" type recording is used in most tape recording equipment employed for business type data processing.

In order to check for transmission garbling it is only necessary that the receiving device check the "one bits" for the proper parity ratio. The device can easily be designed to sound an alarm when the check is wrong. The systems are usually designed either to stop the transmitting device and sound an alarm or automatically to retransmit the garbled section. The early devices mostly just stopped and sounded an alarm. The trend of transmission system design now appears to be toward automatic retransmission. If the transmission is not received correctly in three tries, for instance, the equipment will then stop and sound an alarm. Adding such features as automatic retransmission

increases equipment costs but may decrease the hold time on transmission lines. It may also reduce the amount of operator supervision required.

Elementary analysis will show that a simple parity check will not detect all types of garbles. If two ones in a character are garbled to produce zeros, the parity remains the same. This is a so-called double error. A variety of schemes are used to increase the effectiveness of parity methods. Adding a longitudinal parity bit in each track at the end of a block of data to re-enforce the transverse character parity bit is an example. In some cases controlled redundancy bits are added to the extent necessary for automatic correction of the most common types of errors. These are termed self-correcting systems. Self correcting systems do not require retransmission to accomplish correction. The extent to which they will be employed in business type data communications has yet to be determined.

On-Line versus Off-Line Transmission

You may sometimes hear the expressions "on-line" and "off-line" transmission used in connection with business type data communication. On-line type operation implies that the data processing equipment is directly connected to the transmission circuits for more or less immediate processing of data as soon as they become available. A good example of an on-line application would be an air line passenger reservation system. In such a system, terminal devices in ticket offices are connected to a central processor by transmission circuits. The customer waits for the processor to provide information on the availability of seat space. The design objective may call for a response in a very few seconds. On-line operations of this type are sometimes referred to as "real-time" operation particularly if processing proceeds without significant time delay.

In an off-line application, information is usually accumulated over a period of time and processed in batches either when convenient or at regular periodic intervals. Payroll processing and stock record updating are representative off-line applications.

Conclusion

The use of electrical communications in business type data processing is growing rapidly. New types of transmission equipment are being developed by manufacturers and offered to customers. The commercial communication companies are striving to adjust their transmission facilities and services to meet the customers' needs. The present period is one of dynamic change. I have not emphasized available hardware in this discussion because of the rapid rate of obsolescence of the present equipment.

The acceptance and general application of standards of character coding and recording formats should open the way for development of compatible communication systems in which each manufacturer's equipment can be applied to the system in the most efficient and economical manner.

INFORMATION RETRIEVAL

E. D. Schmitz

Ellsworth D. Schmitz is ADP Performance Analyst, Data Processing Systems Division, Navy Management Office. He has a Bachelor's Degree and graduate work at the University of Chicago, with a major in Public Administration. His eleven years in the Navy Department have been devoted primarily to Data Processing and Management Analysis.

The Problem

The subject of information retrieval, that is, the business of finding needed information in a "store" of information, is in some respects as old as recorded history. Yet, in other respects this subject is as new as tomorrow's headlines.

Information retrieval is old, in that for hundreds of years historians and scholars, and even managers, have tried to learn the present and predict the future by reading, researching, or studying the past in the form of the recorded word.

But as populations grew the number of recorded words grew, man's recorded knowledge grew, the need to communicate knowledge grew, and the media for recording and communicating knowledge grew. The problem of finding valuable recorded knowledge increased and multiplied.

Masses of historical and technical data have been accumulated in various formats -- books, newspapers, magazines, journals, professional publications, and scientific and technical papers. If all this recorded knowledge were placed in one stack, it would certainly reach around the moon. For all intents and purposes much of this information, including some of importance, is lost. It cannot be located.

In a sense, we are faced with a paradox: the more information we gather, the more we seem to want. Each breakthrough in knowledge promises and demands further breakthrough. Studying any real problem today, technical, scientific, or managerial, becomes a real task of locating and studying relevant existing knowledge to see what has been successfully or unsuccessfully tried.

Unless new ways of recording, storing, and, most important, finding information, are developed, the whole effort toward increased knowledge could grind to a halt.

Basically, the problem is to select from masses of data, much of which is irrelevant, that comparatively small segment which is relevant and will help to set up an hypothesis or determine avenues of potential success. It isn't exactly like trying to find a needle in a haystack. It is more like trying to find a particular spear of hay in a haystack.

For lack of means for finding particular useful knowledge or information in the large accumulation of information, much research effort is wasted in developing solutions that have already been found and documented. Duplication and the waste of valuable research time are just two results.

In sum, there is a problem, old, but growing ever more rapidly, in which we are faced with an ever larger fund of knowledge, coupled with an increasing inability to make use of it, because we can't find it.

Past Efforts

In the past, and in many cases, in the present, efforts to locate information have been largely limited to the use of relatively simple classification -- indexing systems. You are all familiar with library card systems, where cards containing title and author are filed alphabetically as a search media. There are also systems for indexing periodicals. We can find some of what we are looking for if we know the title or author, or name of the magazine, or other document; that is, if we know that what we are looking for exists. The problem really mounts when we aren't sure what, if anything, has been recorded, or in what media, or by whom, about the subject in which we are interested.

More Recent Attempts

Recently, there have been attempts to begin to solve this kind of problem by considering the potential of machines for mechanization of indexes, documents, or both. In retrospect, it's somewhat amazing that searchers did not sooner and in much greater numbers seriously explore mechanization as part of an answer. Now there is a large amount of effort, much of it since 1950, to focus attention on the problem, both from the standpoint of developing better indexes or finding media, and using and developing mechanical means for searching. The effort is scattered and various, with different groups concentrating on various parts of the problem.

Examples

Much work at improving indexes is based on some variation of one concept. There are various names for this concept. Regardless of the

names, instead of using a fixed classification system with items indexed by author and/or title and general subject, a set of terms are devised which in sum represent the subject matter of a given library of information. In simple outline, this is the system:

A set of descriptive terms is devised, and each term is entered on a card.

A document is received and read.

An abstract or synopsis of the document is prepared.

The reader determines what descriptive term(s) from the index or vocabulary describe what the document is about.

The document is given a serial number, and filed.

Cards, each containing one of the descriptive terms, are pulled from file.

The document number is written on each of these cards.

The abstract is written on a card bearing the serial number of the document.

The cards with descriptive words are re-filed alphabetically.

The abstract cards are filed in numerical sequence.

In attempting to find information on a given subject, the searcher follows this sequence:

Suppose we have a number of descriptive terms, among which are "metal," "heat," and "expansion". Each of these will be on a separate card, and the cards will be filed alphabetically.

Suppose documents with the numbers as indicated below are in a file which deal with the subjects of "heat", "metal" and "expansion".

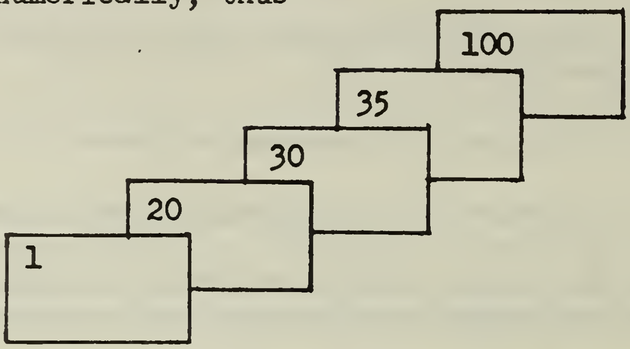
DESCRIPTORS:	HEAT	METAL	EXPANSION
Document Nos.	1, 20, 30, 35, 100	1, 20, 30, 100	.30,100

This means that the subject of <u>heat</u> is dealt with in documents Nos. 1, 20, 30, 35, and 100.

The subject of <u>metal</u> is dealt with in documents Nos. 1, 20, 30, and 100.

The subject of expansion is dealt with in documents Nos. 30 and 100.

For each of these documents there will be an abstract card, and these cards will be filed numerically, thus --



Now comes the searcher who wants to know, for example whether there are documents in file which deal with the effect of heat on the expansion of metal. So,

He pulls out the cards containing the descriptive terms "expansion," "heat," and "metal."

He compares the document numbers listed on the cards.

Any document numbers,and only those, which appear on all three cards will deal with the subject of "Effect of Heat on Expansion of Metal." In our example, documents numbered 30 and 100 satisfy the criteria.

The searcher then goes to the abstract file to pull cards with these common document numbers.

He reads the abstract. If the abstract looks promising, he gets the document and then

He refiles the cards.

This oversimplified system, with variations in degree of sophistication, terminology, and detail, gives a rough picture of the direction of many attempts to improve indexing.

Mechanization

You have probably already seen a potential for mechanization of at least part of this kind of system. The indexes could be on punched cards, or both indexes and abstracts could be on paper or magnetic tapes. Obviously, libraries with many documents and with large numbers of terms would make the matching of document numbers a laborious job.

So the indexes, the abstracts, and/or the documents themselves could be recorded in a machine language for machine matching and searching.

Some Kinds of Semi-mechanical and Mechanical Devices

There is time to describe only a few of some devices which represent the kinds of things being tried and being used to mechanize or semi-mechanize retrieval systems. One simple method involves the use of cards with holes around the edge. In the center of the card, there is space for titles, abstracts, or bibliographic information. Usually there is one card for each document, and notches around the edge of the card represent the subject matter contents of the document. A long needle is used to select the desired cards. This system is economical and useful for systems with a small number of documents and a small number of descriptive terms.

Another simple system involves the use of cards similar to EAM cards. There is one card for each descriptive term. Holes are punched with a simple hand punching device similar to those used by conductors to cancel tickets. These holes represent the number(s) of the document(s) which contain information on the subject indicated by the descriptive term. Cards with terms describing the desired subject matter are pulled from the file, stacked together, and held up to a light. Those areas where light shines through indicate the number(s) of the document(s) which contain information on the selected subject.

A variation on this punched card theme involves the use of cards which are much larger than the standard EAM card. The principle is the same, but after cards containing the desired descriptive terms have been selected, they are stacked and placed on a table or box with a light source underneath. Again, the areas where light shines through indicate the number(s) of the document(s) which contain information sought. This system allows for the identification of more documents per descriptive term (or per card) and is suitable for larger libraries.

Of course, instead of searching punched cards manually, various sorting machines can be used.

Microfilm aperture cards are also used to mechanize the index to documents, combined with a miniaturized picture of the document itself. The principle is similar to those described above. The index is punched on an EAM card, and a small hole is cut in the card into which a piece of microfilm can be placed. Once the sought for terms are selected by hand or machine, the microfilm can be enlarged on a reader for the searcher to determine the value of the document. If desired, the microfilm can be reproduced as hard copy.

Several photographic systems have been developed in attempts to combine the index with the documents. These are generally high priced.

One such system uses small bits of microfilm which contain microcopies of documents, together with their identification in code. These bits of microfilm, each about the size of a postage stamp, can hold up to 12 documents with codes. They are stored on sticks not unlike skewers, each skewer holding 2,000 bits of microfilm. Desired documents are selected by machines. The searcher can view the selected bits of microfilm on a reader, and have those he wants reproduced.

A developing variation on this theme records the documents and index codes on rolls of microfilm rather than on pieces of microfilm. Machines search the film and make microcopies of the desired documents on another roll of film. These selected documents can then be viewed on a reader, and if "hard copy" is needed, a button is pushed, causing a fast copy to be made.

Finally, electronic data processing machines can do the machine operations necessary for mechanized searching systems. With digital computers, the following things can be "mechanized": The Index File, the terms describing the desired subject matter, the file of abstracts, and the printing of results, (i.e., the document(s) and the abstract(s).)

This brief skip through gives some idea of the direction and kinds of effort to improve the search for information. The efforts can be put in several classes. There are those systems which are low in cost, limited in what they can do, and in the volume they can handle, primarily manually, and without giving radical improvement.

There are other systems which are reasonable in price, reasonable in what they can do and in volume handled, more mechanized, and offering fair improvement over past systems. They use conventional equipment.

There are other systems which give much greater speed of search, can handle large volumes of documents, combine documents and indexes, and are usually too costly for most libraries or stores of information.

Summary

What can be said of all efforts to date is that they seem to deal with only part of the problem of information documentation and retrieval. They concentrate mainly on mechanization of the search through an index or file, and on "coordinate indexing" of documents. Although these are steps in the right direction, there is far more to the problem. For example, regardless of the system used, some person must devise an effective and economical set of descriptive terms for each "library." Somebody must choose what documents to receive, index, and file. Somebody must read each document, prepare an abstract, and determine what terms properly describe the document. These are costly and time consuming actions, and upon their accuracy depends the usefulness of the mechanized

search. The searcher must understand and properly use the chosen system. The problems of semantics, "noise," and "misses" remain.

In effect most effort has been concentrated and money spent on mechanizing that which is easily mechanized, and not on those areas most in need of improvement. But librarians, historians, equipment manufacturers, scientists, technicians, mathematicians, and a host of "Information Retrieval Specialists" are spending time on trying to define the problems more clearly, to develop vocabularies and languages, to determine optimum mixes of file size, detail of indexes, and costs of avoiding "misses" and eliminating "noise," and to develop means of mechanizing the "reading" and indexing of documents and the preparation of abstracts at reasonable price.

These are the promising and challenging areas. This is a vital subject, worthy of and receiving increasing attention. Although most people will not be able to afford the more elaborate photographic and electronic devices being devised for machine handling of searches, and although dramatic solutions may be five years, ten years, or longer away, all managers can seriously study and define their information retrieval problem, and keep up with developments, so that present and future "answers" will be recognized as they appear.

THE ELECTRONIC DIGITAL COMPUTER -- CONCEPTS, CHARACTERISTICS, MAJOR COMPONENTS AND THEIR FUNCTIONS, AND MACHINE COMMANDS 1/

Samuel N. Alexander

Samuel N. Alexander is Chief of the Data Processing Systems Division of the National Bureau of Standards. His field includes the application of digital techniques to data processing, information storage and retrieval, and the application of combined analog and digital techniques to automatic instrumentation and dynamics systems control. He was formerly Chief of the Electronics Computer Laboratory on development of digital computer technology and overall supervisor of design and construction of SEAC and DUSEAC.

To build up a little perspective, I'd like to draw upon some applications of the use of this equipment which will help to classify them. There's nothing hard and fast about these classifications, but I think they'll help in visualizing the type of work you will find the equipment doing.

The initial application of this class of equipment was for making calculations, and from this comes the name Computer. Almost concurrently with this use, though, was the application to some form of data processing -- the outstanding example being the early experiments at the Census Bureau. A third class of application consisted in carrying out particular kinds of data processing, an operation I'd like to call "Data Manipulation," and which tends to be a mixture of calculation and something else that isn't normally associated with calculations. That's why I used the word "manipulation." Suppose you have some data, and you want to rearange the format of the fields. You do not perform calculations. You may invert fields, you may abbreviate fields, you may insert other data. This class of manipulation of the data can go to very extensive lengths, and I'd like to give an example or two a little later. And, finally, there is a fourth class that I call "Information Retrieval." I use the term "Information" rather than "data" at this point because in most cases input to the machine has grammatical structure of some sort.

1/ This paper is an edited version of a tape recording of the lecture by Dr. Alexander. The Editor of this volume is responsible for the editing.

REGULATIONS

HOURS

HOURS - 0830 to 1730 Monday through Friday. No books charged after 1715.

LOAN PERIODS

BOOKS - may be borrowed for 2 weeks. Renewals are granted on request for 3 additional 2-week periods provided no request for the same book is on file. CALL Ext. 55413.

PERIODICALS - may be borrowed for 3 days. To renew, CALL Ext. 55413.

LOST OR DAMAGED BOOKS AND PERIODICALS

Borrower is responsible for safe return of library materials and will replace any lost or damaged books or periodicals.

INTERLIBRARY LOAN SERVICE

Publications not in the Army Library may be borrowed through this service from other libraries for official use only. They must be returned promptly.

TELEPHONE NUMBERS

Circulation	55413
General Reference	74301
Legal Reference	52957
Military Documents	55535
Periodicals	73897
Interlibrary Loan	73279

BOOK MARK

THIS BOOK IS CHARGED TO *YOU* AND *MUST* BE RETURNED ON OR BEFORE DATE STAMPED BELOW:

~~27 NOV 1967~~

~~8 JAN 1968~~

30 JAN 1968

Prompt return of this book is essential for the efficient operation of the Army Library. If necessary to retain beyond the date due, you *MUST* obtain a renewal in accordance with the provisions on reverse of this book mark.

Your cooperation will enable us to provide better service at less cost to the Government.

THE ARMY LIBRARY
ROOM 1A-518, THE PENTAGON

AGAZ FORM 718
1 OCT 58

Even in data processing, we tend to have something that is remotely akin to grammar. Thus, a catalog entry, there is meaning to the arrangement, which is the old idea of fields, but there isn't much in the way of clear-cut adjective type of modifiers -- or clear-cut action verbs -- but there is an elementary beginning in cataloging entries. In the Information Retrieval field, the desire is actually to handle information that appears in the form of brief statements. If you could really do what you would like to do, you would hope some day to enter an abstract of an article or a journal and ask the computer to answer questions from that abstract. We are some distance from this.

A field very close to both Information Retrieval and Data Manipulation is the field of Translation of Languages -- probably the most outstanding example of data manipulation. In this case, the computer does not attempt to answer any questions about the content of sentences translated; it's trying simply to transform the material from one standard format to another. Unfortunately, this isn't as clean-cut as it sounds because it's perfectly possible to make a statement that is grammatically correct, and yet be completely non-sensical. Therefore, you'll have to be very careful that in this transformation from the form of one language to the form of another you don't lose the sense. The outstanding gag in this field is the case in which the machine was asked to translate "out of sight, out of mind" and came back with "blind idiot". So, for what it's worth, I group applications into calculation procedures, processing procedures, manipulating procedures, and retrieval procedures, and I use "information" in order to emphasize the fact that we deal with things that have structure; or in more fancy words with material that has syntactical character.

You'll remember in studying grammar taking a sentence apart and identifying its pieces. They have order and relationship. Incidentally, the use of syntax helps pack more information per number of symbols than any other device that man has invented. So, it is also a means of compressing the information into smaller space, if you know the rules for syntax. For example, when I write this symbol (123) on the board, you all instinctively interpret it as the number "one hundred twenty three." That's because we have agreed on some rules, some conventions that are almost instinctive, in which position has meaning. On the other hand, to show a variation, I could have been writing in base 5 -- in which case this would not correspond to 123 (that is, 1 times 100, plus 2 times 10, plus 3 times 1). It would have meant 1 times 25, plus 2 times 5, plus 3 times 1. So, you must know what convention is being applied at the moment. Now, that is the most elementary kind of meaning in terms of order; position has value.

There are other schemes we use when we build up to a fancy scheme which is a language. Thus, there is quite a different meaning to the "2" when put as a superscript as against the "2" when used as a subscript.

Position has meaning. Then there are other schemes for getting even more meaning, depending on order and relationships, to get all the way up to the pinnacle, represented by natural language with all its rules and nuances.

Another characteristic of the Information Retrieval type of work which will be discussed in more detail in a later lecture is the fact that, as distinct from the usual data processing task, where you work into a file that is almost complete (or as complete as it's worth your while), the Information Retrieval problem almost invariably works into a file that is known to be incomplete or has entries known to be out of date. You must use this file in spite of this deficiency. The best examples I can offer here is the FBI files, or police files in general. They're reasonably complete, but there is a lot of missing information, in comparison with the personnel files of your agency, in which the information is freely and voluntarily given and there is an incentive on the part of the staff to see that their personnel files are correct and up to date. How do you work with a file that is known to be incomplete, with elements out of date? Also, how do you ask questions about the file when you know for sure that the way in which questions will be asked very likely will not correspond to the way the file has been arranged? Or, again you may have the situation in which all you know about the individual is some particular peculiarity of his makeup, some physical characteristic, or something about where he was last seen, and not his name or his serial number, or that he came from Joliet, Illinois. I draw these distinctions because they have a great deal to do with the way you use the equipment and the kind of equipment balance you ask for to do the job. So much for this method of classifying jobs.

The original field of calculation was what touched the development off and the great preoccupation initially was in improving the arithmetic unit of these machines. In particular, the ability to multiply much faster was a very important characteristic at the outset. The reason was that in most scientific calculations, the class of problems that had not been adequately treated were those in which multiplication and division occurred with great frequency. Because of this, the figure of merit of the machine (the number of units of finished work per dollar of capital investment or per dollar of operating cost) was closely tied to how fast it could multiply. To most of us, this is a quite secondary characteristic.

Many applications are much more concerned with handling large volumes of data, and this was precisely the point at which the Census problem put pressure for a shift in the technical development. The requirements for the Census of 1950 put a severe technical demand on the original Univac with respect to the quantity of data that could be handled on a tape. A magnetic tape was chosen as the most likely candidate for meeting the needs of this class of work. Whereas with respect to the computational tasks of that day, most of the volume could be handled by adding a few magnetic drums to the machine in addition tothe working members, not so

for the Census problem or for the first life insurance problem. The equivalent of a file was the essential innovation that had to be made a practical working entity.

I would like to put in perspective for you the development of the tapes on the Univac I. I would say that the cost of developing those tapes -- and making them acceptable for use -- compares in cost to the development of the rest of the computer. Most people don't give enough recognition to this point. Another way of looking at it is, the equipment -- electrical, mechanical, and electronic -- necessary to control the input-output system consisting of tapes and typewriters and keys for printers, etc., (that is, the things that have to do with preparing material to go into the machine, holding and controlling the movements of the tapes, writing back onto the tapes, and taking the tapes off the machine and turning out hard copy) accounts for close to half of the complete investment. So, it is pretty evident that calling this kind of assembly a computer is only a half truth.

The situation that lead to the development of the magnetic tape mechanism and the form which it finally took was heavily weighted by the fact that the insides of the machine were still considerably faster than the tapes. We had gone in one jump from a punch card of conventional size to a piece of magnetic tape of about this size (perhaps 3/8" x 1") which corresponded functionally to the punch card. With respect to data reading rates, we had gone from about 100 characters a second reading off of cards to 10,000 characters a second in one great big jump. And yet the electronic equipment attached to these tapes was still much faster. If you didn't carefully plan your work, or the organization of the machine, the inside of the machine most of the time would be waiting for the tapes. This lead to two kinds of endeavors. The first was a technical endeavor to improve the speed of the tapes (today the standard tape unit is 20,000 characters per second and there are new ones on the market just coming into use which run in the range of 60,000 to 90,000 characters per second and a few of the manufacturers are holding out prospects of a hundred to 125,000 characters per second). The other effort was an intellectual one rather than a technical one. It was an effort to plan the work so that every time you took data off the tapes, into the machine, you really wrung them dry, i.e. performed as many possible operations on those data as your overall system required. This is the reason why simple transliteration of a punch card procedure doesn't do a very good job. The punch card procedures inherently recognize that the interpreting mechanisms inside the punch card device are in reasonable balance with the rate at which cards can be fed. You very seldom see a phenomenon in a punch card operation in which the cards wait very long for the insides of the machine.

In the original Census task, the tapes never came to rest; the tapes were just going all the time but the arithmetic unit was operating a very small fraction of the time. They gradually reorganized their work,

however, until in their present operation you can see the tape coming to rest, just momentarily. They put a lot of effort into trying to make it come out reasonably well balanced. In fact, they did a little bit too much of a job. (This is a little aside. During the first two months of operation,the tapes were coming to rest and starting up at such a fantastic rate that they were wearing out the clutches on the tape unit. In the course of one week they were getting what the manufacturer had normally considered would be a year of life of the clutch. This has, incidentally, been solved by changing the material on the clutch.) I just wanted to point out that they went along the path of trying to organize the work so that you could extract as much as possible. Notice the change in economy and the change in your system philosophy as a result of this equipment consideration. It cost you the most expensive operation you have in getting the information on the tape -- I'll talk about that a little later.

The next most expensive unit operation is getting data from tape into the computer and therefore you want to do as much as you possibly can on that material while it's in the computer before you release it and put it out. I've gotten a little ahead of my story. I mentioned we were up in the range of 100,000 characters per second as the frontier today. Unfortunately, the fellows designing the electronics in the computer have done better than that and so the system is out of balance, internally, all over again. Early calculation rates of 1000 additions, subtractions, and comparisons a second have been shoved up to a hundred thousand a second. It's no real strain today to put in equipment that will do a hundred thousand additions, subtractions, or comparisons per second, if you want to pay the tariff. So the tapes have gone up by factors of ten -- from ten thousand to a hundred thousand -- and operations in the machine have gone up by factors of a hundred. So, the imbalance is actually worse now than it was when we started out. This is leading to efforts to try to offset this. And we're going backwards in principle. Remember, I pointed out that in the early days the fellows doing calculations found that they could handle the bulk with which they were working on several drums attached to the machine in addition to the main high-speed memory, whatever it was. The present trend is to supply lots of drums to serve the function that we normally think of serving with two or three tape units. The progression today is tending to take on the following form: (drawing in board).

This is a stack of discs or a battery of drums, depending on which manufacturer's product you're talking about. One manufacturer is proposing a larger number of drums, others are proposing stacks of discs, and I think the tendency will be toward discs because the economics favor them. I'll explain more about them in a moment.

These in turn, feed whatever is the high speed memory. And under this arrangement, the tapes, no matter how fast they are, have become the loading device for the system. This has become the intermediate or

overflow store and is the unit from which operations equivalent to sorting are done. And the high-speed memory, now, serves the same purpose as before, but because of some of the problems introduced by trying to live with a working volume that's not more than two or three tapes, the requirement for this memory has gone up. When we started out two years ago, four thousand words were considered a nice comfortable memory size, but many of the manufacturers today are talking 32,000 and going up to a hundred thousand words of memory available at the customer's option. This volume of memory and this peculiar restriction are interrelated. Now, let me go into this just a little bit. What you would like is something that gets around the time it takes to find the particular items of data that you want, not limited by the running time on the tapes. You want something more nearly random in its access capability.

The high-speed memory is completely random. You ask for a particular piece of information and it points directly to it and takes it out. It isn't completely random by digit in most cases, but it's completely random by words. You can find any one word, and if you want a piece of a word you can bring it out in the arithmetic unit and take it apart.

For example, consider a book. This is random by page and it's random by line on the page, but sequential within the page. In order to speed this up, they're trying for access cycle of a tenth second. Notice I'm drawing a distinction between access time and data rate. The data rate is the rate at which the numbers come pouring out at you when you finally arrive at the point you want. The access cycle is the time it takes to reach the number you want. These two numbers are the same for the high-speed memory -- the core-type memory. They differ for this thing in the sense that they're trying for a tenth second arbitrary access and a data rate use to the order of a hundred thousand characters a second. We are able to achieve something of the same sort for the tape unit with respect to data rate. Let's take one that you can buy off the shelf -- on the order of thirty thousand characters per second. The ninety is also used, but the access cycle can be -- depending on how many tape units you want to invest in, which in turn determines how long a reel of tape you work with -- anything from one minute (a short tape) to five minutes. If you try to use a grand-daddy size roll, it can run as high as five minutes from one end to another, depending on how much you want to pay for the mechanism.

You're trying to live with the fact that you can't quite predict exactly what you are going to want next. If you had a good clear idea of exactly what you want next, then you could provide enough lead time. It's like buying material for a shop. If you know exactly what you need when you need it, you can build up a lead time procedure and bring it in off of the tapes. Since you can't quite, you bring much of the material eventually into a warehouse in your own organization, and then you draw from this warehouse. And this, essentially, corresponds to the working material right at the consumption point to the extent that you know

roughly, statistically, how things are going to be called, for you can arrange for staging from here into here. And you're finding the kind of special cicuitry built into these devices that permits you essentially to provide a lead time. You can tell this unit to begin delivery of so many items and then go about your internal processing, and you know that these items will accumulate and be available when you want them. This is like a drip pail unit with a trip on the pail to pour it real quickly.

In a couple of other machines, there are automatic devices for staging the information automatically, from here into here, and then taking it again quickly from here into here. Do you follow my point on this one? You can leave in a word for so many units and you can tell the machine to put it here, and then when the machine is ready for the next batch of work it can call for a transfer out of here. The worst that can happen is a tenth second delay plus a recount which is usually small compared to the tenth seconds. Okay, now how can it get to the tenth seconds? Again, this is because of mechanical time for turning a wheel. And, incidentally, there isn't too much you can do about some of this. A reel of tape isn't a rigid solid body in the same sense that a wheel is. If you accelerate a reel of tape too sharply the turns will slip, one with respect to the other. The reel holds itself together by the friction between turns and while it's perfectly possible to design a motor clutch arrangement that could put a lot of torque on the shaft, the reel of tape won't take it. You'll have to go to all kinds of clever things to keep the reel under good snug tension to avoid this effect. So, even though you would be willing to pay the price, you're getting near the end of what's durable. There are some proposals to design the reel to hold the tape in such a way that you can start them and stop them faster. But this is only going to change it by a small percentage. You may be able to double the thing, but we're talking about imbalances. So, this is not going really to solve the problem. Also, the other problem is trying to run the tape faster. It's possible to speed the tape along 20, 30, or 50 feet a second. We've actually run magnetic tapes successfully at 30 feet a second. But you get into all kinds of nasty troubles. It sucks a film of air in between the tape and the head, and try as you will, you can't get that tape snug against the head unless you drill a hole down through the head and provide vacuum.. Thus you are playing with complications. So, we try to be a little smarter and some of the smartness leads to complexity in the machine.

Let's look at this in principle. Here's a magnetic drum. The usual idea of a drum is a head at every track. And if you really wanted to be wasteful, you put a set of electronics at every head. It turns out that this is unnecessary, for the turning time of the drum is of the order of a hundredth of a second. A hundredth of a second means a hundred revolutions a second; that's six thousand revolutions a minute. That's a fairly fast drum. It's quite easy to switch, even with relays, from one circuit to another in a hundredth of a second, and with electronics, it's no trick at all. So, you can have a common set of electronics that is switched from head to head. This is the usual procedure.

The heads are one of the most expensive parts of this assembly. At a very small cost rotational processions developed for other applications in industry can be bought. But the magnetic heads are of the order of \$25 apiece at the low end of the scale, probably more like \$100 apiece for the magnetic reading head. So, this is one of the large elements of expense. But you must go to that expense if you want to stay within this hundredth of a second access time. If, however, you're willing to go down to a tenth of a second, it's possible to make a trade and this is what most of them are trying to do. Just put on one magnetic head and think of a real strong stainless steel band tape (parallel to the axis of the drum) and a drive motor of some sort at each end. Whenever you want to get to a particular track, you energize these motors. They yank this stainless steel tape one way or the other to locate the head on the right track, which is equivalent to turning to the right page of a book. This designates the page and there's a detent mechanism to make sure that it stops exactly in the right place. This is one scheme for getting at this situation which gives fairly good performance. You can steal a good deal of this technology (and modify it) from the anti-aircraft gun devices. In fact, they use hydraulic servos or similar devices right off the anti-aircraft guns. However, this takes up too much space.

Notice that for a drum, the active part is the outside surface; the inside is waste space. It's just needed to hold the surface in the right place. So, the general temptation is to go to a hollow cylinder. Now you can get at both the inside of the drum, and they share a common drive system. It's just trying to be a little more mechanically ingenius. You pay for it, for it's much harder to make a plate that holds its position, for the end without the bearing can wobble. There are a whole lot of problems. You don't get something for nothing. You can't really be sure that this surface is exactly where you want it to be; whereas, by simple careful machining the exterior surface is exactly where you'll want it to be. So, for this sort of device, the reading head in addition to being able to move in and out, has to have a means (which is usually a little air jet stream) that sucks it nearly up against the plate -- but not quite. It works somewhat on the principle of a child's toy consisting of a pipe and a ping pong ball. You blow on the pipe and it will lift the ball only so far. No matter how hard you blow, you can't blow the ball out. If you keep blowing, you can turn the thing upside down and the ball will stay right in position. Now that air-bearing type of principle is what's usually used in recognition of the fact that this surface may give some trouble. Owing to gravity, it's going to droop a little as you go out to the edge. We're talking about one part in a thousand, or maybe as small as one in ten thousand change in clearance. So, they make use of air-bearing tricks.

Now, whether or not you really make a good trade between the different mechanical features, you'll have to listen to the sales talk. You don't have to listen to the complaints of the maintenance people in order to get a real value judgment on it. These are two quite different ways

of approaching the problem. They have advantages and disadvantages. Where the balance point is, I think the government's pocketbook will have to find out. We're probably the biggest users of this equipment and we'll probably develop the biggest background of knowledge on this. All right, so much for the mechanics of this.

Now, we've got a scheme to go from here to here to here (pointing). A request from here can be filled in a tenth of a second. The general design objective, and I don't know whether it is being met, is to have your information requirement organized so that you can ask for the elements in sequential steps. I want to start at the edge here and I want to go progressively. I want to take the next twenty tracks, one right after the other. The hope is that you can do that in a half of a tenth. In other words, they are the order of a twentieth of a second. There has been some talk of being able to do it a little bit better if you know you're going to inch in one right after the other. My own prediction is that this is going out, that the real way out of this rather unhappy mess of too much machinery to maintain is to learn how to make magnetic heads in mass production at such a low cost that you will put magnetic heads in in every position on the drum. Go back to the drum again -- the original drum -- and use electronic switches. Take your initial capital cost and buy it back in operating cost -- operating reliability -- or buy it back in reduced maintenance cost. At the moment, no one has found it worth his while to design automatic production procedures for making magnetic heads, but it's certainly no worse than automatic grinding of optical lenses, which we learned how to do in the last war. So much for my estimate for the future in this area.

Now all of this talk about several kinds of step-ups in memory and organization in the machine in order to make use of the extra capacity that's possible with a faster memory and a faster arithmetic logic obviously has its roots in the questions "Who wants to work that fast?", "Who's got that much volume?", "What jobs are worth all this extra trouble?". Well, this is a fair question. There are some jobs that really are worth it. As before in the first cycle of events, the second major cycle of events is coming from the pressure from weapons development. The Livermore Laboratory of the Atomic Energy Commission started this cycle and several years ago they contracted for a machine called the Larc -- Livermore Atomic Research Computer. The price tag to the David Taylor Model Basin is about four million dollars. I think the market price is going to be more like five and a quarter. This is an indication of rising cost and an underestimate of the complexity of doing this design. The first of these has been delivered. The second one has been advertised and they are trying to take orders for additional ones now called the Stretch. This one was designed for the Los Alamos Laboratories of the Atomic Energy Commission. Again, for weapons work, this one was also seriously considered. I believe the main justification for the Navy was in connection with power reactor calculations. Some of the data

processing jobs have enough similiarity so that the final character of the Larc is such that it ought to have considerable utility for data processing jobs. In fact one firm did a quick analysis of the Larc versus the mixture of 704's and 705's that his corporation had planned to install over the next several years; and they concluded that if you could make full use of all of the features of the Larc and if you could handle the management of interlacing the workloads -- not the technology of interlacing the workloads -- that a couple of these machines would replace half a dozen installations at a considerable saving in cost. In other words they thought that one of these Larcs could absorb the workload of three 704's to three 705's or something of that sort. However, it poses a serious managerial job of getting everybody lined up meeting everybody's priority.

The Stretch is not yet delivered. It represents considerable improvement in speed over the Larc and is the one that has gone the farthest in the development of the disc memory as against the drum or trolley heads. Both of them are providing quite sizable stores. The question that I raised before and I left hanging was "Who among us might conceivably have this kind of workload?" Well, one of the ways of taking advantage of these developments is consolidated workloads. This is the option. The other thing is that we are moving toward complicated tasks that are running off the end of the available machines. The reason the first round of tasks fit so nicely on the first computers is that we had given up long ago and had settled the automation of relatively simple tasks because that's all the punch cards should do and we have a body of knowledge and problems which are mostly of that sort. Let me give you an example, so you might not think I'm just publicizing the machine's work. We have a cooperative task with Federal Communications Commission and in that job, one of the tasks is to determine interference if you ask for a new channel assignment. To determine the possible interference of requested assignments with all the registered assignments is a brute on any machine. So much so that you have to give ground very sharply on what refinements you know could be put in. The thing was quite complicated, and I think the administrative procedure you had to live with consists of giving an applicant a nominal power assignment, and after the station is in, revising the power that he can actually use based on field measurements. Now, this is roughly the way the procedure goes. It seems possible that with the more detailed knowledge that we have today of radio propagation we could determine the interference to a much better limit, except that the computation is quite elaborate.

This leads us to the next problem which goes to show you how the technology intertwines with procedures. It would seem ridiculous to make everybody in the United States that wants to apply for a license go through this calculation all for himself. And it leads to the suggestion that the Federal Communication Commission ought to carry out these calculations in full detail and make them available for every possible channel that is presently available. Then the applicant's job

would be to address himself to the remainder of the task. Is he willing to put up that big antenna? Can he buy that much real estate? Is he willing to conform to the other requirements, the economic regulatory requirements? This particular one could be solved once and for all in full and published. Essentially the ideal would be that the FCC, instead of putting the burden on the applicant, would provide a shopping list. Here are the channels open and here are the requirements to utilize these channels. Anybody interested?

The areas in which we think the machines will have utility unless somebody becomes very, very ingenious, is in the information retrieval area where one is stuck with the fact that he must work with a lot of incompleteness in the file and where the situation involves many interpretations and one inserts into the machine as many rules for screening among these interpretations as can be formulated. I can't give an example quite in this form, but again let me revert to one of the Census. Census had tackled a problem a few years ago of materials that come in by questionnaires through the mail. These questionnaires were usually on commercial activity and industrial capacities and things of this sort. These questionnaires must be filled out, under the law; but there's nothing in the law that gives any real way of making sure the questionnaires are filled out carefully. If you can prove that the man is deliberately falsifying, you can go after him. But if his secretary gets the date in the column that corresponds to the number of employees, accidentally, (this is a reasonable error), it's up to you to catch this before it goes into your statistics. If you ask the man to report his product in tons per whatever is the standard for that time period, and he keeps his records in thousands of pounds and he forgets to convert them to tons when he fills out the questionnaire, you can't very well do much to him, except if you catch it, to call his attention to it on the hope that he won't do it again. You can list a whole grouping of things in typing a form -- adjacent digits transposed, all of the headaches that occur in copying from one record to another. You don't want to bias your statistics; you'd like to catch these. Well, the effort along this line was called input edit, and they set up simple screening rules as, if the sum of the labor cost plus material plus rent and so forth is ten times the cost of the sales value of products,then question the form. You would think that you wouldn't catch anything by that, yet they would catch a reasonable number of improperly filled out forms which were of this class -- wrong units or digits transposed in the typing. So that such a simple screening rule as the fact that the value of the product was only a tenth of all the cost were examples of the kind of things they got. So they began putting tighter and tighter screening rules for having the machine reject; and they found that they were getting something approaching thirty per cent of the forms with some trivial errors in them. And this obviously meant actual manual handling of thirty per cent of the cases. So then they had to go into the next cycle which was to devise reasonable restitution rules;

recognize the adjacent digit transposition by comparing this year's production against last year's production or against the average for the industry, or getting the productivity of that particular establishment and comparing it to the productivity of the rest of the industry. And so they were able to find many ways of correcting the data in the file. Since they were doing statistics they were free to do one more thing which is not usually given to those who have to handle financial records, and that is, where the thing was obviously inconsistent and no restitution rules seemed likely to apply, if the item concerned was less than a certain fraction of the total for the industry, strike out that entry and put in the average for the industry on that title, and keep all the others. By keeping a running control on this, they could prevent biasing the statistics. In other words, the machine again is doing the kind of thing that they probably would direct a clerk to do in a manual operation.

Well, now if you have followed this far, you'll find that this can get to be quite involved and it can become quite a hog for the machine-time. And yet, it's the type of thing that if you don't do in the machine, you're going to do outside with people. Now it's a race between two things. First can the people do it cheaper and more promptly? As the unit cost per calculation comes down the productivity of the machine is better than the productivity of the earlier machine. Secondly, in many cases the kind of talent in a clerical staff necessary to do this work is beginning to vanish, and you haven't much choice. If you don't build it into machine procedures, you're going more and more to have to do without this kind of service -- this kind of cleaning up in your statistics. So, in some cases, this is the only serious alternative you have, and this can build up work-loads very seriously. Let me finish on this point by saying that I'm convinced that as we go through the easy type of payroll personnel records, all those well-organized nicely-behaved jobs, as we get more to the heart of the activity of the agencies, you're going to find that in automating these operations, they're going to be surprisingly complex. And in order to handle this complexity, you're going to drive in the direction, maybe not of these big fellows, but to machines like the 7090 and like what Census is now using -- the 1105 Remington Rand and the Philco 2000. I think these machines are going to be the order of the day, because we're going to learn how to put more of the things that look alive -- items of judgment -- which are really not judgment in the real sense, but enough intelligence to know where to look for the regulations. If that is the job, what are the guidelines to tell you where in the regulations to find your answer, that is potentially susceptible to machine operations. These tend to be quite complicated when you describe them in the baby language that the computer uses; and I offer to you this example of the input edit as a very simple earthy example of where it occurs.

I'd like to close; I see I've not talked about any where near all of the items I wanted to cover: I want to talk on the problem of obsolescence that seems to be posed by new machines coming. The rate of change seems to be such that this field won't settle down for another ten years.

Everytime we turn around somebody comes up with an improved idea. An improved component and the pressure of more complicated workloads leads to a more cleverly organized machine. The field is not settling down at the present time. What do we mean then, under these circumstances, by obsolescence? Well, the word obsolescence, I think, has to be treated with two entirely different meanings, depending on whether you are coming to the field for the first time or whether you have some equipment and are considering replacement. Quite different standards apply.

Obviously, if you are coming into the field and you have no prior investment, then a machine that's transistorized and has all the other glamorous features ought to be considered in terms of the durability of the investment before you have to consider a replacement. On the other hand, if you have a machine in place, the economics of obsolescence now are in terms of "If I make the change, and I give up some capital goods, and make a larger payment, at what rate do the savings in operations take care of this difference before I actually begin to break even?" And you'll find that this isn't so very attractive. Air conditioning and floor space rent are quite cheap by comparison to the difference in the cost of the machines. Furthermore, transistorization only reduces the size of that part of the machine that has to do with computing. The discs are just as big, tapes are just as big, the consoles are just as big; tape bends, the tape records, the off-line printers -- all these things are just as big as before. My guess is that with the most complete transistorization (cutting floor space in half would be a reasonable objective) it will take you a long time for the savings in rent to buy back the difference in cost of the transistorized machine against the machine you have. Therefore, you've got to get it back out of higher productivity. So, it isn't clear-cut when you come to change. If there were a well established second-hand market for machines in government, you would really simplify the problem of choosing the fancy new machines against some of these quite competent clunkers. The best example again I can give you is that Census intended to give up both of their Univac I's and then decided to keep them.

A part of this is the conventional implication of obsolescence. Usually people think of obsolete machinery, in addition to being uneconomic to operate, as being worn out, or almost worn out. This concept, which comes from engines and automobiles and so forth, isn't really applicable here. In order to keep a computer at a level of reliability to make it useful for your work, you have to keep it in tip-top shape all the time. Remember the famous axe -- "it had ten new handles and seven new heads, but it was still the same axe." Of course the computer isn't quite physically the same, but you can maintain the machine so that it's still almost as good as it was when you got it. Otherwise you can't get a good day's work out of it. Now, this idea is also a rather tricky one, so you'll have to watch out. These machines don't wear out in that sense. The nearest thing to wearing out that I can offer you is the wiring and some of the material inside the machine gets brittle and in the course of

maintenance work, people may break insulation or wires, etc. -- this is the nearest thing to the engine being worn out.

Another point with respect to obsolescence or the difference between the modern shiny machines and the older ones is that the transistor -- which is usually the hallmark of the new and modern machine -- has improved that part of the machine which was already good enough. The arithmetic control and so forth is not a major source of lost time. It is probably the third or fourth ranking source of lost time. And adding transistors to this part of the machine just makes the part that was good, very much better. So, you don't get a real improvement in available time compared to what you would think if you compared the reliability of a vacuum tube against the reliability of a transistor. Vacuum tube machines are already good enough. The problems are out at the tapes, the printers, and things of this sort. Also, the transistorized machines are not without a requirement for ventilation and sometimes air-conditioning. I think the newer ones have reached the point there they don't require air-conditioning, but they still require a ventilating system. Because, in order to make them compact they cram the stuff together and although there is less heat coming out, it has a harder time to get out. Also, the transistors are much more sensitive to heat than the vacuum tubes. In fact, the vacuum tube is the least sensitive to heat; it gives out heat. You know, like the story, "I don't have ulcers, I give them." I don't know whether the vacuum tube is in that category. It's the other parts that deteriorate owing to the vacuum tubes' heat. In the transistorized machine, the transistor is the most heat sensitive element in the whole assembly. So, it's not all good. Now, let me see -- I don't want to knock my profession. It is a quite definite improvement and with the passage of time it's going to be even better. Pretty soon the time will come when the electronics of the machine will probably be sealed almost the way the mechanisms of some of the electrical clocks are sealed. You just won't bother to repair them. When the day comes that you no longer use them, you throw them away. (A question was asked from the audience)

Well, tapes do have the unfortunate characteristic of wearing out with use -- it depends on how hard use you give them. The common coated tapes actually have only about two ten-thousandths or three ten-thousandths of an inch of material layed onto the mylar. Therefore, after a number of passes over the head, the first loss is usually to wear out the head. It grinds away. Secondly, the tape finally begins to wear out, and one of the things you've got to watch out for is uses for the tape in which you see-saw back and forth in the same region. There was one nice example in which somebody had a routine that did this, and he found that he wore through the tape about every two or three days and so he had a procedure for copying this part a few inches farther down on the tape. This is not an extreme case, but he actually wore all the way through the coating in a few days of running.

The other source of obsolescence is, if your tape doesn't wear out on you, the changing tape base from acetate to mylar is permitting putting

more footage on a reel. It is also believed that the mylar will stand storage over periods of time much better, but the question about long-term storage has to do with the bond between the active coating and the base. It isn't quite clear that the active coating bonds to the mylar with as much surety as it bonds to the old acetate. An unanswered problem with respect to these things is in archiving material.

The other one is that the ability to store on the tapes is going up, as I indicated, from ten thousand to twenty thousand to thirty, sixty, and ninety thousand characters. I believe that the difference between sixty and ninety is not in packing it tighter, but in a wider tape. I think we've got to be careful in that distinction. This is not quite obsolescence now; there's a difference rather than an obsolescence. Tape that had uniformity, freedom from imperfections and things of this sort, that were satisfactory, put down ten thousand to the inch. I'm sure it would be good enough if you took the same stock and tried to use it on the devices that would let it down at sixty thousand to the inch. I'm speculating on this, I'm not sure. I think I'm in the right direction. So the quality required of the base tape is higher for some of these newer applications.

However, the volume of tape being used these days is rising so rapidly that the unit cost of the base material is falling, so that the major cost to the user is not the cost of the tape, the physical material -- the coating -- but the testing and screening and whatever other operations the manufacturer puts it through before he delivers it to you with a guarantee. A good example of a spread here is, I think Remington Rand which, I think, charges between $50 and $70 a reel for the tape after it comes through all the checking and the warranty procedures that they would use in tapes for delivery to Census. Census can buy the same material from the manufacturer with his tests and screens, which are fairly complete, at $30 a reel. And then they must do a few more checks before they actually put it into use. But they use a machine now to check it out. So there's another difference.

Another problem which is bothersome to a lot of us is the fact that people are ingenious in the ways of squeezing a little more out of a tape. So the tapes come in somewhat different widths, different packing, a number of heads, and the way the information is written on them. We're getting into a regular Tower of Babel with respect to conversion. This is being offset slightly by the fact that nearly all of the manufacturers who are making tape mechanisms independently of complete computer systems are trying to offer at least the Remington Rand and the IBM configuration as standard. On the other hand, I believe the RCA tapes are different. Certainly the Minneapolis-Honeywell tapes are different. So that you've still got a problem. We've got a serious Tower of Babel sort of problem on how to convert one to the other inexpensively if the government is to give all the suppliers a reasonable opportunity to market their products. I think we should accept whatever inefficiency there is in

this conversion in order to give the major producers that kind of competition, for it will bring back into the government far more savings than the cost of the converters. I think this is a problem of organizing.

I'll close with this: One of the proposals is that in each major government area there be a very complete conversion unit that serves as a kind of inter-agency service bureau, so that there's an inexpensive way of making these conversions in each large government community area, and conversion does not become a dominant consideration in choice of equipment. I'd subscribe very heartily to this because those of you who can remember the punch card field know that the dominance of IBM in the punch card business was an unhappy situation for many of us as compared to the present situation where we have lively competition. Furthermore the tape mechanism is one of the most sensitive points in the efficiency in the overall system -- if you don't go to this disc arrangement. And therefore it behooves us to let the natural competitive forces hammer out the best possible tape design, and you would hope that in the long run the best design will work its way into the machines as they settle down. This will mean a lot in your operating efficiency.

The same is true to keep up the competition with respect to these discs. The thing that's heartened me most is that there is a small independent concern that's designing these discs and supplying them for sale to the minor producers of computing machines, so that they can stay in this rather fantastically expensive competition with the big fellows.

And I'd like to close with this one note that to date most of the real innovations have not come from the big fellows. Their advertising gives you the impression that it does, but it isn't true. The real innovations and the push for change have come from the smaller fellows. So we have quite a stake in keeping them going, and when you worry about your equipment choices, please think of this. Try to be a little bit public-spirited in your choice if you possibly can.

PART 3

ADP SYSTEMS PROGRAMMING

BASIC COMPUTER PROGRAMMING AND FLOW CHARTING

M. H. Schwartz

Mr. Schwartz has been at the Federal Reserve Board for the past 10 years and has specialized in statistical operations and electronic computing for 6 years. He has developed a variety of training courses in electronic computers with considerable attention to use by persons whose primary interest is in subject matter analysis. He has an A.B. Degree from the University of Illinois and has completed substantial graduate work in the fields of Economics and Statistics.

The objective of today's seminar is a discussion of the elements of instructing the machine (coding and programming), machine logic, and flow charting. This seminar will, I hope, help you understand the character of programming "at the machine level" and will help prepare you for Dr. Hopper's seminar on automatic programming, or programming "at the object level."

But first I should like to review with you those basic facts about electronic digital computer components and their organization that are highly relevant to programming. Past seminars have shown the desirability of such a review at this point. Getting into the new world of computers does take some digesting!

The digital computer is fundamentally an arithmetic device, so that the arithmetic component is a natural starting place for review.

The arithmetic component, or "unit" as it is more frequently called, consists of electronic circuitry and components capable of performing basic arithmetic operations one at a time in serial fashion. It derives much of its usefulness to us because the individual operations are performed at such lightning speed that a very long string of arithmetic is accomplished in a relatively short time. When you think of the arithmetic unit as solving a complicated equation, think of it as doing it in a way that differs from more familiar methods mainly in its speed. The computer does not simply jump to the result; it gets there by a sequence of operations such as you or I might perform on a desk calculator. Indeed, the arithmetic unit of a computer is helpfully visualized as a very special speedy kind of desk calculator.

The arithmetic unit is algebraic, i.e., it pays attention to the signs of the data. The addition of two positive numbers will yield a positive result and the subtraction of a large positive number from a small positive number will yield a negative result. In its treatment of

the signs of raw data, then, the arithmetic unit of the electronic computer behaves in the everyday way we do when we do arithmetic, and signs will be given to results in the same everyday manner. This is of course true not only of addition and subtraction, but of multiplication and division. When the machine divides the number "positive 10" by the number "negative 2", it will yield as a result the number "negative 5."

Except when you and I perform the most trivial arithmetic steps with a desk calculator, we sit down to a task at the machine with pencil and paper. One major purpose of the paper is as a work sheet. Such a work sheet could contain columns or rows filled with raw data, and additional blank columns or rows into which we could post intermediate and final results as we stepped through a job. The computer also needs a work sheet for the same purposes. The computer's work sheet is, as you might guess, another package of electronic circuitry and materials and it is generally called "storage" -- because it "stores" raw data at the outset of a problem and then it "stores" intermediate results and final results as a problem is solved. You have no doubt heard about a computer's "memory." This is a rather overimaginative word for computer storage devices -- at least at this season of our lives.

The internal organization of computer storage is generally likened, for people getting started in the field of computer applications, to the electronic equivalent of the familiar post office or messenger office pigeon box. Whereas each "cell" of the post office pigeon box will hold a number of materials of diverse kinds, however, each "cell" of computer storage will hold one and only one piece of information at a time. Fixed-word-length machines hold a number of a specified size in each cell, while variable-word-length machines hold a single character in each cell.

Let us focus on the fixed-word-length machine for the time being. When we talk about such a machine, we are talking about a machine that always handles increments of information of a standard length according to a particular design. The increment is a computer "word." The IBM-650 and Burrough's 220 machines handle 10-digit numbers or words, while the Univac I handles 12-digit numbers. In the case of the IBM and Burroughs equipment, there is an additional position for the algebraic sign of each number or word. In machines like the Univac I, however, one of the 12-digit positions must be used as a sign if the data may be both positive and negative during the course of a problem. In a 10-digit fixed-word-length machine, numbers that are short to you and me, like "1", or "27", or "3,018", are represented by 10-digit numbers filled with leading zeros -- 0000000001, 0000000027, 0000003018. Numbers that exceed the 10-digit capacity are carried around as two separate 10-digit numbers that are linked together for arithmetic operations. This, incidentally, is termed "double precision" because the basic capacity of the machine for handling individual numbers is doubled in the process of linking two words together. Triple, quadruple, and other orders of precision are feasible for the unusual cases where such precision is needed.

Each cell of the post office pigeon box represents a particular geographic entity, such as a street, or a city, or a state, and each cell is generally identified by its contents. One talks about the "New York" cell, or the "downtown" cell, or the "out-of-state" cell. In computer storage, however, each cell is called a location and each is referred to by what is termed an "address." Cells are generally addressed sequentially. The first cell will be number 1, the second number 2, and so forth, up to the capacity of the storage unit. Thus we say that a particular computer has a storage capacity of "N" locations or "N" words whose addresses run from 1 through N.

Before I tell you how the programmer works with the storage unit, and I almost got into that discussion at this point, let me tell you about the three remaining and readily distinguishable major components of a present-day electronic computer. First of all, we need to get information to the computer. We need to fit the raw data into the storage unit so that we can put the arithmetic unit to work on it. Getting information into the computer is done by the machine's "input" unit. The simplest electronic computing machines have keyboard inputs, rather like a desk calculator. Most equipment, however, can digest information at much faster speeds than any human being could achieve by manual input. Punched cards comprise the most common form of automatic input.

Each 80-column punched card holds up to 80 characters of information punched as holes in the card. The characters may be numbers, or letters, or punctuation. Today most computers in service will input cards at the rate of 200 per minute -- a rate of 80 x 200 or 16,000 digits per minute. Never machines will handle cards at 800 per minute and even faster. You can see how card speeds run way ahead of manual operations. As each card is "inputted," the holes in the card are sensed by electromechanical or optical equipment, translated to the language of the computer as described for you by Dr. Alexander, and routed to designated locations in storage.

Magnetic tape is a faster input medium than punched cards, faster because the information is packed more tightly along.the tape and because the tape itself can be moved and scanned at greater velocities than can cards. In part, these advantages stem from the fact that information is recorded in magnetic tape as magnetic impulses rather than as holes; and in part these advantages stem from the fact that the tape is a continuous stream that does not involve the separate handling that must be given to each card during an input operation. Whereas tape can be read by many popular computers today at speeds of 15,000 digits per <u>second</u> -- as compared with digits per <u>minute</u> using punched cards -- some manufacturers have announced speeds close to 100,000 digits per second for tomorrow's machines. And still greater speeds are just around the corner.

Once data are "inputted" and the desired work accomplished, results must be outputted. Broadly speaking, there are two kinds of output -- humanly sensible documents and machine language documents. The humanly sensible output is normally furnished by a printer either on-line, i.e.,

fed to the computer and printed during the course of the operation, or off-line, i.e., by printing from a machine language document prepared on-line, such as punched cards or magnetic tape. Machine language outputs are prepared either for off-line translation or as inputs to a computer for another stage of a process or as input at a future date for summarizing, updating, auditing, or reconciliation purposes.

Machine language output speeds are similar to input speeds, except that cards can not be punched as fast as they can be read. Output printing speeds -- a phenomenon we can watch with our eyes and appreciate directly -- have reached literally fantastic rates. A speed of six hundred lines of printing per minute, each line consisting of up to 120 characters, was only a short while ago one of the special features of the largest, most costly computer systems costing a million dollars or more to buy and $20,000 to $25,000 per month to rent. Now it is possible to rent a computer for $3,000 or $4,000 per month, or less, including a printer of 600 lines per minute capacity. Much faster speeds are available with newer, larger machines. Indeed, one hears now of "page printers" as well as "line printers", and of printing electronically without type and without moving parts.

The fifth major component of an electronic computer system ties together the four we have already reviewed. This component is called the "control" unit. The function of the control unit is, in effect, to supervise the execution of the desired work. The control unit "sees" to it that, when the time comes to read a card, the input does read a card; the control unit "sees" to it that the arithmetic operations performed upon the data contained in the card or in storage are properly executed; the control unit moves data in and out of the storage unit as required; and when the time comes to output a result, the control unit in effect "sees" to it that a card is punched, a line is printed, or a magnetic tape block is written.

With this review of machine components behind us, we can now turn to the elements of instructing the machine. "Instructing" is the word that is used to describe in a generic way how the machine is harnessed by the programmer. The operations of the machine are performed in accordance with a program. A program consists of a sequence of instructions to the machine that perform the particular objective of the moment. Each objective requires its own program. But each program consists of a particular concatenation of the same fundamental instructions which the machine is capable of following.

There are several classes of instructions to an electronic computer system. The basic class consists of arithmetic instructions. Other classes of instructions pertain to input/output, that is getting information in and getting results out; logic, that is, "decision making," or, more strictly, performing this or that particular sequence of operations as will be explained in further detail later; and a varying group of miscellaneous "supporting" operations having to do with scaling the decimal point, moving data in storage, and other special functions.

I believe that the fundamental characteristics of a machine instruction have already been presented to you. An instruction to a computer is a grouping of letters and numbers, or numbers only, that cause the machine to perform a particular operation in accordance with the circuitry of the machine. Let me illustrate with a rather far-fetched analogy. When you turn the start key on your automobile you know this sets into motion a surprisingly complex series of operations. A heavy flow of current is fed by the battery to the starter motor. The starter motor engages the flywheel of the engine and spins it. The flywheel is connected to the crankshaft and the spinning of the flywheel rotates the crankshaft. Rotating the crankshaft does several things. The pistons move up and down within the cylinders; gas is pumped to the carburetor and drawn into the cylinders; voltage is furnished to the spark plugs so that they will emit sparks and ignite the gas; and so on.

The computer is, of course, a very different kind of machine. The fundamental difference is that it responds not to mechanical events like the turning of a key or the pressure of a pedal, but rather it responds to numbers and letters, or numbers only, fed to its control system. To one machine, the number 15 might mean add, the number 10 might mean subtract, the number 19 might mean divide, the number 14 might mean multiply. To other machines, the letter A could be used to mean add, S to mean subtract, M to mean multiply, and D to mean divide. C could refer to card handling operations, T to tape, and so on. Each model of machine has its own "code", but all units of the same model have the same code.

Now, an instruction consists of the code, or signal, for an operation together with whatever additional information is required for the code to be effective. An instruction to add, standing in a vacuum, is meaningless. Add what? It is clear that the instruction to the machine must also indicate the "with what" and "to what" the machine is to perform the coded operation. In the case of arithmetic, the operation is generally performed in connection with data in storage. The instruction, therefore, also specifies the address (or addresses) in storage of the data involved.

The details of arithmetic instructions vary among machines and arithmetic instructions can be categorized in various ways. One very broad way depends upon whether the instruction makes specific reference to the arithmetic unit -- or "accumulator" as the arithmetic unit is generally termed in programming. "Single" address machines perform arithmetic with the contents of one datum address per instruction and each arithmetic operation in such a machine contains a specific reference to the accumulator. "Two" and "three" address machines perform arithmetic in connection with two or three data addresses per instruction and instructions do not refer specifically to the accumulator. There are advantages and disadvantages to each of these forms of instructions, depending upon the kind of project involved and the tastes and habits of the programmer.

Examples of the instructions required for the addition of two numbers are as follows:

A. One address machine.

1. Clear the accumulator and add in the contents of a specified location in storage.

2. Add to the contents of the accumulator the contents of a specified location.

3. "Store" (or "copy") the contents of the accumulator in a specified location in storage.

If the numbers to be added were in locations 101 and 102, and the programmer wished to put the sum of the two into location 301, his instructions would be

	Code	Address	
1.	65	101	Clear and add the contents of 101
2.	15	102	Add in the contents of 102
3.	20	301	Store the sum in 301.

B. Two address machine.

1. Add the contents of one location to the contents of another and replace the contents of the second location by the sum.

Code	Address	Address	
6	101	102	Add the contents of 101 and 102 and place the sum in 102.

C. Three address machines.

1. Add the contents of one location to the contents of another location and store the sum in a third location. In machine language, with "11" meaning "add", such an instruction might look like this:

Code	Address	Address	Address
11	101	102	301

In the two-address machine, the result of each arithmetic operation wipes out one of the two data, so that if the programmer wished to

preserve all the data for future use in the same program, he would need to use special copy instructions for locating the data in additional cells in storage before the arithmetic was performed.

The three-address machine has certain advantages for arithmetic operations, which normally involve more than one location by the very nature of arithmetic. For some kinds of arithmetic, the advantage is lost, however; and for many nonarithmetic operations, the three-address system can become wasteful. But the three-address system is often the easiest to understand, so I shall stay with the three-address machine for the balance of this seminar.

The four elementary arithmetic operations are add, subtract, multiply, divide. Let us denote the three addresses of a three-address instruction as the A, B, and C addresses, respectively. Then the four elementry arithmetic operations would work as follows:

ADD (A) + (B) to C	The sum of the contents of the A and B location is placed into the C location.
SUBTRACT (A) - (B) to C	The difference between the contents of the A and B locations is placed into the C location.
MULTIPLY (A) x (B) to C	The product of the contents of the A and B locations is placed into the C location.
DIVIDE (A) ÷ (B) to C	The quotient of the contents of the A and B locations is placed into the C location.

Suppose I had the quantities X in 101, Y in 102, Z in 103, and R in 104 and I wanted to compute the following equation, placing Q into location 105. Suppose 11 were the code for add, 12 the code for subtract, 13 the code for multiply, and 14 the code for divide.

$$Q = \frac{(X + Y)(Z)}{R}$$

My little program would appear as:

Code	A	B	C	
11	101	102	105	(Compute X + Y)
13	105	103	105	(Multiply sum by Z)
14	105	104	105	(Divide product by R)

Note that location 105 is used as a temporary storage for the intermediate results as well as for storage of the final result.

With the specified data and instructions in the machine, when I press the "program start" switch, the machine executes each of the three instructions in sequence, one at a time. If that is all I wanted the machine to do, my fourth instruction would have been simply to bring the computer to a halt. If the code for halt were 01, my fourth instruction would have been:

Code	A	B	C
01	000	000	000

Note the use of zeros in the three unused address positions of the halt instruction.

By this time many of you may be wondering, where are these instructions in the machine? Well, interestingly enough, they too are in storage, right along with the data. The machine can not literally tell them apart, i.e., it does not know data from instructions -- they are both groups of numbers, or numbers and letters. But when the "program start" switch is actuated, the computer begins by executing the instruction in location 1; it then executes the instruction in location 2; then 3; then 4; and so on through the storage -- in consecutive sequence. So if we put our instructions into the low numbered locations, and our data in locations beyond the last used for instructions, we shall be all right. If, however, we inadvertently caused the machine to place the result of some operation into a location in the midst of our instructions -- well, when the computer got to that location in the usual sequence, it would "think" it had an instruction. If the code were valid, i.e., if the two left-hand digits of the number happened to represent a code, the machine would execute what it took to be an instruction. If the two left-hand digits did not represent a valid code, the computer would halt.

This peculiarity of the electronic computer, whereby instructions may look like data, data may look like instructions, and both are in the same storage unit, turns out to provide enormous power, as we shall soon see. But for the moment, please try to visualize both data and instructions in separate sets of locations in the storage unit.

Given the program -- or sequence of instructions -- in the machine, how do data get in? (The program itself gets in by a process known as "program loading," whereby each instruction is copied from a punched card into a sequential storage location by a method which we simply take for granted.) This calls for the use of an input-output instruction. The detailed working of an input-output instruction depends upon the characteristics of the particular machine, but broadly speaking the end effects are the same. There usually are separate codes for each operation -- read a card, punch a card, write a block of magnetic tape, read

a block of magnetic tape, print a line, eject the paper one sheet, and so on. A simple example might look like this:

LOCATION	INSTRUCTION	
001	21 901 000 000	Read a card into sequential locations beginning with 901.
002	11 901 903 801	Add the first and third factors from the card and place the sum in 801.
003	11 902 904 802	Subtract the fourth factor from the second factor and place the difference in 802.
004	22 801 000 000	Punch a card containing these results, i.e., containing the contents of the sequential locations beginning with 801.
005	01 000 000 000	Halt.

A real-life program would look just like our example, except, of course, it would contain a much longer and more involved sequence of instructions. In addition to input-output and arithmetic operations, a real life program is apt to contain both miscellaneous operations and "logical" operations. The two most common miscellaneous operations have to do with lining up the decimal points (and rounding off large numbers) and with copying data from place to place within the storage unit. Examples might be:

31 321 002 000	Shift the contents of 321 two places to the left; i.e., insert two zeros at the right end.
33 422 003 000	Round off the contents of 422 by 3 places; i.e., express the number in location 422 in thousands, properly rounded.
71 208 408 000	Copy the contents of 208 to 408 (at the end of the operation, the contents of 208 will be undisturbed; the previous contents of 408 will be destroyed and the present contents of 408 will be identical with those of 208).

It is the "logic" of the computer that you will find particularly interesting, however, and so I should like to turn in that direction at this point. We can do little more than be illustrative in the time we

have, and I feel sure that you will gain more by a detailed illustration of some logic than by a rundown of the complete repertoire of the logical codes of a modern computer.

Suppose we had a project like a payroll, or an inventory, or a statistical analysis -- whereby we wished to perform the same set of operations on a large batch of inputs. How do we get the machine to execute the same program over and over again and still preserve working space in storage? After all, we would soon use up the largest storage unit if we had to duplicate the program in sets of storage locations for each identical input. And we certainly do not want to spend expensive machine time loading the program separately for each input.

This takes us to our first example of computer logic -- the "unconditional transfer of control". You will remember that the instructions are located in the low numbered locations in storage and that the computer executes them one at a time in consecutive sequence, location by location. A transfer of control is an instruction that causes the machine to interrupt its automatic sequence and to begin a new sequence with the instruction located in a designated location. An unconditional transfer of control is the basic kind of logical instruction.

Suppose a particular project took 108 instructions with each input. The contents of location 109 could in such a case contain an instruction that causes the machine to interrupt its sequence and arbitrarily return to location 1. There it would find the first instruction and would once again go through the 108 instructions. Each time the machine came to location 109, it would turn back to 1. The last two instructions might look like these:

LOCATION	INSTRUCTION	
108	22 801 000 000	Punch an output card with results.
109	41 001 000 000	Transfer control to location 001; i.e., once again begin the sequence of instructions located in 001.

This is called "cyclic programming" because it causes the computer to "cycle" around and around, through the same set of instructions, in a way analagous to the rotation of a cycling wheel. The process is also frequently known as "looping."

Thus a particular program may be executed on an indefinitely large size input and may generate a correspondingly large size output. The "logical" provision that a programmer needs to make to set up a repetitious use of a particular program is the inclusion of the unconditional transfer of control at the end of the program loop. It is not

necessary that the transfer of control be to the initial location, of course; it could be to any location consistent with the program. Sometimes the programmer uses the unconditional transfer to jump ahead rather than to return.

The unconditional transfer of control is the beginning of an advanced level of automaticity in computer operations remotely controlled by programming. There are two additional classes of computer logic which I should like to introduce to you that greatly expand this new level of automaticity and which will go far, I believe, in helping you understand the potentials of programming. These two classes of logic are called "counting and controlling" and "address modification."

We need one more kind of computer instruction in order to perform these last kinds of operations. This kind of instruction is the "conditional transfer of control" -- i.e., a transfer of control that is neither arbitrary nor mandatory but rather is one that either is performed or is not performed depending upon the status of some location in storage or upon the status of certain devices. For example, there is an instruction to "transfer on a nonzero condition." This causes the machine to examine the contents of a specified location. If the contents of the location are not all zeros, the machine transfers control to a specified location. If the contents of the location examined happen all to be zero, the machine would not transfer control, i.e., it would simply continue on to the next instruction in automatic sequence.

Let operation code 42 mean the following to the computer: examine the contents of the location specified by the A address of instruction; if the contents are nonzero, transfer control to the location specified by the B address; but if the contents are zero, continue on in normal automatic sequence. Thus, if in location 032 the computer encountered the instruction,

Code	A	B	C
42	101	001	000,

this would work as follows: if the contents of location 101 were nonzero, the computer would transfer control to location 001; but if the contents were zero, the computer would continue on and would next execute the instruction in location 033.

It is frequently desirable for a given cyclic program, or "loop" as it is called in computer jargon, to be performed a particular number of times. For example, there may be 52 cards containing weekly data for a particular year. The program to be performed may require (1) identical operations using the data for each week and then (2) summary operations upon the results of the operations performed on the detailed data. The summary operations could not be performed until all 52 cards had been

processed. Each pass through the loop would involve reading a card, performing specified operations, and transferring back to the beginning of the loop.

How to get out of this loop after it has been performed the desired number of times? This is effected by a Conditional transfer of control. One technique is for the program at the outset to contain in some location the number of desired iterations of the loop. Each time the loop is performed, counting is achieved by subtracting a "1" from the location. The program then commands the computer to test the contents of the location, using the transfer nonzero instruction. If the contents are nonzero, i.e., 52, 51, 50, 49 ..., the computer is to transfer control back to the beginning of the loop. If however, the contents of the location have been reduced to zero, the computer is not to transfer back, but is now to continue in normal sequence at which point the final summary operations will begin.

Suppose the group of instructions -- or loop -- required to perform the detailed work on each of the 52 cards were contained in locations 001 through 023. Then the "counting and controlling" operations that will cause the loop to be performed exactly 52 times would appear in locations 024 and 025 as shown below, and at the outset of the whole program, locations 101 and 102 would contain the values "52" and "1."

Location	Instruction			
	Code	A	B	C
024	12	101	102	101
025	42	101	001	000

After the loop in locations 001 through 023 is executed the first time, the instruction in location 024 is executed the first time. This instruction says "subtract the contents of 102 from the contents of 101 and replace the original contents of 101 by this result." The effect of instruction would be to subtract 1 from 52 and to place the new value 51 into location 101. This is called "counting down." The computer then performs the instruction in location 025, which causes it to test the value of 101 for nonzero or zero. Since it will find the value 51, which is of course nonzero, the computer will transfer control back to location 001 where, in turn, it will begin to process the second of the 52 cards.

After each pass through the loop, the contents of location 101 will be consecutively counted down from 52. Passes through the loop will be continued until the contents of location 101 are counted down to zero. It is after the instruction in location 024 is performed the 52nd time

that the contents of 101 will be counted down to zero. At that point, the 52 cards will have been processed, and the test instruction in location 025 this time will not transfer back to the beginning of the loop. Rather, this instruction will now find zeros in location 101 and will now permit the computer to continue in normal sequence to location 026, where the instructions for performing the summary operations would be encountered.

Beyond counting and controlling lies the logical class of operations known as "address modification." Suppose that there were two sets of data, each of 150 items, in locations 201 through 350 and locations 401 through 550. The addition of the elements of these two sets of data, or in computer terminology, the elements of these two arrays, could be performed with results going into locations 601 through 750, by a sequence of add instructions written by the programmer as follows:

Location	Contents			
	Code	A	B	C
001	11	201	401	601
002	11	202	402	602
003	11	203	403	603
004	11	204	404	604
*	*			*
076	11	276	476	676
077	11	277	477	677
*	*			*
0149	11	349	549	749
0150	11	350	550	750

This solution to the problem would require that the programmer write 150 ADD instructions.

An alternative would be to write a program containing two classes of operations, one to perform the desired arithmetic on the data, the other to perform operations on the program. This latter may be visualized as an extension of the "counting and controlling" studied earlier. The process, called "address modification", is a powerful tool in programming repetitious operations on data in sequences of locations.

Since the instructions are stored in the machine as numbers, the program has the same access to them as to any other number within the machine. Moreover, the program can perform the same kind of operations on the numbers representing instructions that it can perform on numbers representing data.

In other words, instructions can be considered as a special type of numerical data and as such they are subject to arithmetic operations. Take the case of the following segment of a program, which includes adding the two arrays.

Location	Code	A	B	C	
001	11	201	401	601	(ADD)
002	11	001	095	001	(ADD)
003	41	000	000	001	(TRANSFER)
*	*			*	
095	00	001	001	001	

When the computer performs the instruction in location 001, it will add the first elements of the two arrays and store the sum as the first element of a new array. It will then perform the instruction in location 002, the execution of which on paper would look as follows:

11 201 401 601	(contents of 001 before)
+00 001 001 001	(contents of 095)
11 202 402 602	(contents of 001 after)

This is, the instruction in location 002 causes the computer to add the contents of location 095 to those of 001 and to store the result in location 001. At the time the computer is executing the instruction in location 002 it does not "know" that the contents of location 001 are an instruction; the computer simply treats 001 as any other address. After performing the addition specified in location 002 the computer thans performs the instruction located in 003, which transfers control back to location 001. There the computer now finds, not 11 201 401 601, the original instruction, but 11 202 402 602 -- which causes it to add the second elements of the arrays. The computer thus performs an instruction which is the result of operations performed by it in executing the program as written by the programmer.

The computer would be in a "loop" in cases like that described above. It would perform the instructions in 001, 002, 003, 001, 002, 003, 001 ... one after the other in a continuous looping sequence, each time through the loop adding the next elements in the two arrays and storing the sums successively in a third array.

Termination of a loop embodying address modification may be effected by the same sort of counting down process used earlier to control 52 passes through a read-compute loop:

Location	Code	A	B	C		
001	11	201	401	601	(ADD)	(Data)
002	11	001	095	001	(ADD)	(Address Modification)
003	12	096	097	096	(SUB)	(Count down)
004	41	000	096	001	(TRN)	(Control)
*	*			*		
095	00	001	001	001		
096	00	000	000	150		
097	00	000	000	001		

Every time the instruction in location 001 is performed, two elements of the arrays will be summed; every time the instruction in location 002 is performed, the instruction in 001 will be modified to handle the next elements in the arrays; every time that the instruction in location 003 is performed, the contents of location 096 are counted down by 1; every time the instruction in location 004 is performed, control is transferred back to location 001 <u>until</u> the entire loop has been performed 150 times. At that point, control will not be transferred back to 001; rather the computer will continue on to the instruction located in 005, in normal sequence.

It has been my intention up to this point to help you grasp both the nature of programming and the potentials of programming for accomplishing certain kinds of work. Let me help you take but one more step along these general lines before I turn to the subject of flow charting. There are other conditional transfer instructions besides that to transfer on a nonzero value. One other key conditional transfer instruction is a test for a minus condition. That is, the machine examines the sign

of the number in the location specified by the A address. If the sign of this number is minus, the machine transfers control to the address specified by the B location; if the sign is positive, the machine continues control in the usual automatic sequential manner.

The best example I can think of to illustrate the use of the test-for-minus instruction in data processing is in inventory accounting. After each issuance of an item from stock, the computer usually computes new balance-on-hand. Suppose that after each such computation the computer then subtracts from the balance on hand (plus amount on order) the minimum amount plus 1 that it is desired to have on hand and on order. If the result is minus, this means that the desired minimum is a larger value than the actual -- which in turn means that an order must be placed for additional quantities of the item. After performing such a test subtraction, the computer would either transfer to a special section in the program to notify the data user of the particular condition or it would continue on to process the next item, depending upon whether or not the item currently being processed was or was not in adequate supply.

So much for programming for the moment. Linked with programming is the subject of flow charting, a topic also scheduled for today's seminar. A flow chart is a graphical representation, step by step, of either a process or a computer program. It is a highly stylized graphical representation in which a limited number of standardized symbols are used, in a particular manner, so that anyone knowing the rules can study the chart and know what it represents. The flow chart is a summary representation and the level of detail is based on the complexity of the project; the greater the complexity, the more detailed the chart may need to be. The flow chart should cover key points, turning points, tests, and checks. Since it serves as a guide or map, it must be accurate at <u>some</u> level of detail. However, it is not necessary, or even desirable, that it contain every possible detail.

The purposes of a flow chart are to aid in planning, to formalize thinking, to focus attention on details of operations, and to bring to light what the project is about. A flow chart is a communications document.

There are two kinds of flow charts -- program flow charts and process flow charts.

The program flow chart lays out the main elements of the computer program by revealing the arithmetic and logical steps to be followed, the kinds of data to be included, and the kinds of output to be prepared. It is a valuable tool; the program flow chart aids in minimizing programming errors, wasted effort, and backtracking for the programmer. It serves as a document of communication and as a basis for discussion among various persons concerned with the program.

Since the program flow chart lays out the dimensions of the whole job, it serves as an indication to the programmer as to whether the job can be done as indicated or if a new flow chart will have to be made; in other words the flow chart helps to indicate whether the proper approach is being taken in the first place.

For those of you who have not had the opportunity to use flow charts for one reason or another in the past, let me present one sample set of computer program flow chart symbols. Remember that there are many sets, none of which is necessarily the best. So long as we know the rules for a particular set, we will be able to handle the symbols. (Sample shown on page 88 following; at the seminar, copies of the sample are to be handed out. At the same time, a sample flow chart will be handed out; it appears on page 89 below.) The object of the sample flow chart we have just passed on to you is to show some major elements of a payroll computation, merely to convey the sense of a flow chart.

The nature of the organization or job decides who is to prepare the program flow chart. In a large organization or on a very large job, the flow chart will very likely be written by the Systems Analyst; in a medium-sized organization it may be a joint effort between the programmer and a person familiar with the particular job at hand; in a small organization or on a small job within a large organization, both the flow chart and the program may be prepared by the programmer.

The program flow chart can be made in advance or after the program has been written. It is made after the program is written if: (1) there was no flow chart to begin with; (2) the advance flow chart was inadequate; (3) the advance flow chart was in effect revised by the program as finally written.

Whether or not a flow chart was prepared in advance, there <u>must always</u> be a flow chart after a program has been run, for the following reasons: (1) Staff changes -- another person must be able to pick up the program and understand it. (The flow chart is keyed back to the program; it will contain some set of codes or numbers or notes referring to the body of the program which will be readily recognized by persons working on the program.) (2) Since programs usually have to be extended, the flow chart is <u>imperative</u> as part of the <u>documentation</u> of a program. <u>In the hands of enterprising management, programs are living things, tending to grow and expand and increase in power and perform more and more services</u>.

The process flow chart is a pictorial representation of the <u>processes</u> required to execute some project. Process flow charts embrace all the steps involved in a process and may include many beyond those performed by the computer.

Included in these processes would be those associated with document preparation, data transcroption to cards and to tape, verification of the

Figure 1.--A Set of Standard Flow Chart Symbols

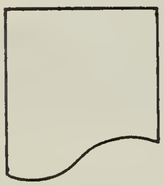
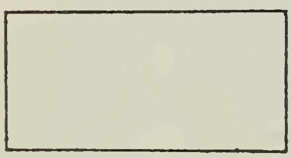
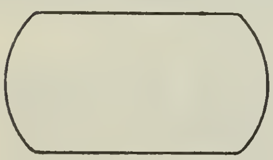
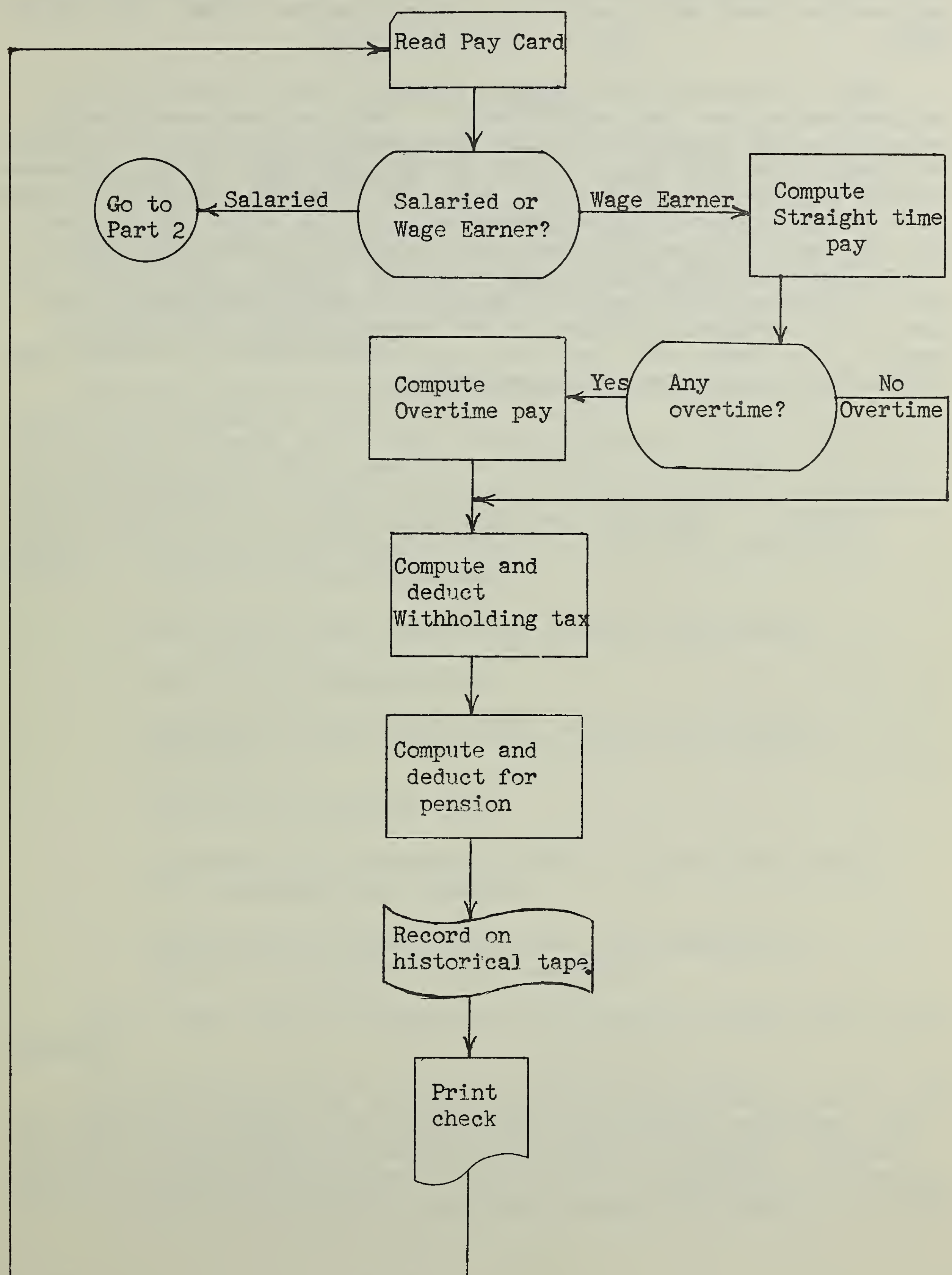
Symbol	Reference	Remarks
	Punched Card	Read a card; punch a card
	Magnetic Tape	Read tape; write tape; rewind tape
	Printer	Print a table, or a check, or a statement
	Console	Operator intervenes at console to take care of some special circumstance
	Perform Arithmetic	Compute amount of interest; Divide by age; Y=(X+R)(Z)
	Perform a Test	End of file? Male or female? Inventory adequate? Loop performed specified number of times?
	Do something to the program	Count down; modify instruction

Figure 2.--Sample Flow Chart -- Elements of a Payroll



accuracy of the transcription, collating operations, computer operations, and finally the disposition made of computer outputs. The computer or program flow chart may or may not be included as one element of a process chart.

Here at the end of our seminar, I should like to bring us back to programming for one final half-minute. I hope you have seen from the material we have covered so far that programming brings great flexibility and power to the data user. It also brings great power to the programmer. Address modification, for example, helps the programmer write his program; it is an operation for the programmer more than it is for the data user. The data user would get the same results whether the programmer wrote a "brute force" program or an elegant "looping" program. My point is that many programming techniques exist mainly to simplify programming. The power of the machine is focused on the program as such. In your next seminar, Mrs. Hopper will guide you to an appreciation of a vast new use of the computer for simplifying programming.

AUTOMATIC CODING -- 1960

Dr. Grace Hopper

Dr. Grace Murray Hopper is Director, Research--Systems and Programming, Remington-Rand Division of Sperry Rand Corporation. Dr. Hopper assisted in programming for the MARK I, II, and III Computers, was a Research Fellow in the Harvard Computation Laboratory, played a leading role in programming the BINAC and UNIVAC systems and in the development of automatic programming, is a national authority on automatic coding, and author of numerous technical papers.

Not very long after the first computers were delivered, it became clear that programming them was about to become a costly bottleneck with respect both to money and to time. More programmers did not alleviate the situation since there seems to be a limit to the number that can be usefully employed on a single application. The solution to the difficulty seemed to lie in more programming per programmer. Automatic coding -- the use of the computer itself to assist the programming effort -- could offer these advantages:

Reduction of time required for original programming

Reduction of debugging time

Reduction of time required for revising and updating programs

Reduction of training time

Increased use of systems analysts in closer relationship with programmers and computer

Re-creation of communications net from management to computer by means of natural languages.

All of these would contribute also to reduction of the cost of programming.

Reduction of the time required for original programming implies that the amount of "coding" to be written shall be reduced, and hence, that some form of short-hand or "pseudo-code" shall be invented. Further, coded and checked-out sections of coding must be expressed in a sensible form in order that they may be stored and incorporated into new routines.

Less coding to be written means fewer opportunities for making mistakes. Thus, debugging time is reduced by the use of previously checked sections of coding and by fewer mistakes in the coding. Since revising and updating can be carried out at the pseudo-code level, and the computer instructed to make necessary changes, program maintenance time is reduced.

Programmer training time is reduced because it is no longer necessary to learn the finest detail of the operation of each and every computer component and instruction. It is only necessary to learn pseudo-codes defining computer operations at a functional level. With a shorter training time, it becomes economical to teach systems analysts as well as programmers to prepare problems for computer operation. As the pseudo-codes grow to correspond to natural languages, the solutions of problems become comprehensible to all concerned, from top management down to the computer itself.

Relative Addressing

In the early days, studies were made of the types and frequencies of the most common coder mistakes. Incorrect addresses of storage locations caused by alien digits, interchanged digits, non-sequential quantities, and errors in transcribing digits had the highest frequency. Therefore, the earliest efforts were directed toward simplification of the addressing techniques.

Consider the problem of adding 100 quantities Y_i to 100 quantities X_i, adding a constant k to each sum and storing the results as Z_i's. First, let the 100 X's be stored at locations 800-899; the 100 Y's at locations 1200-1299; k at 585; the 100 Z's at 1529-1628; and the instructions starting at location 115. The program then might be (in the instruction code of no known computer):

115	Get	800	X_0
116	Add	1200	$+Y_0$
117	Add	585	$+k$
118	Park at	1529	$=Z_0$ to Z_0
119	Get	115	(Get 800)
120	Add	586	(Get 801) (a "one" stored at 586)
121	Compare with	587	(Get 899) ("Get 899" stored at 587)
122	If greater go to	XXX	If greater, exit from routine

123	Park at	115	If less, send (Get 801) to 115
124	Get	116	(Add 1200)
125	Add	586	(Add 1201)
126	Park at	116	Send (Add 1201) to 116
127	Get	118	(Park at 1529)
128	Add	586	(Park at 1530)
129	Park at	118	Send (Park at 1530) to 118
130	Jump to	115	

This is a "fixed subroutine", it is not only written in computer code, but it refers to fixed storage locations.

Suppose, now, the assumption is made that the X's lie from A00 to A99; the Y's from B00 to B99; the Z's from C00 to C99; the k constants start at K00; and the instructions at M00. The subroutine can then be written as:

M00	Get	A00	X_0
M01	Add	B00	$+Y_0$
M02	Add	K00	$+k$
M03	Park at	C00	$=Z_0$ to Z_0
M04	Get	M00	(Get A00)
M05	Add	K01	(Get A01)
M06	Compare with	K02	(Get A99)
M07	If greater go to	XXX	If greater, go to exit
M08	Park at	M00	If less, send (Get A01) to M00
M09	Get	M01	(Add B00)
M10	Add	K01	(Add B01)
M11	Park at	M01	Send (Add B01) to M01

M12	Get	M03	(Park at C00)
M13	Add	K01	(Park at C01)
M14	Park at	M03	Send (Park at C01) to M03
M15	Jump to	M00	Jump to M00.

The subroutine itself is now coded relative to an arbitrary starting point M00. The input-output areas are referred to as the A, B, and C areas, while the constants are stored in a K-area.

If A=800 B=1200 C=1529
K=585 M=115,

are added in place of the letters in the relative coding, the subroutine returns to its original specific position. Another set of values for A, B, C, K, and M will position the coded section in some other program.

Several programmers can now work on a problem simultaneously. Error-free sections can be written because they are short and the elapsed time from problem definition to production running on the computer can be reduced. A "library" can be set up on tape of relatively coded sections of program and these can be integrated into new programs. As the power of the assembly systems is increased, and as their libraries grow, subroutines, or pieces of relative coding, are called from the library by "call-words" or "pseudo-codes" and only the allocation of storage remains as the task of the programmer.

The next step in the development of automatic coding is the "compiler". A compiler is a routine which accepts a "pseudo-code" or a natural language, interprets the pseudo-code, selects subroutines and generative subroutines from a library, supervises the generation of subroutines, allocates data and instruction storage, assembles a complete program, and prepares a report on the program.

Subroutines

Subroutines were recognized by the very first large digital computer. MARK I at Harvard, as designed by Professor Howard H. Aiken, contained wired-in subroutines for cos x, log x, and exp x. Since these functions were wired into the machine, they, of necessity, offered complete generality: cos x accepted any value of x in radians, between minus infinity and plus infinity; it delivered the function, cos x, accurate to 21 decimal places. Almost never was such a degree of accuracy required. If x represents the angle of the roll of a ship, it is surely known that x is less than $\pi/2$. If the problem data is submitted with five decimal places, 21 places are not required in the cosine function. The wired-in subroutines proved to be too general. Very soon, subroutines were programmed, such as one to compute sin x to 8 decimal places for $x < \pi/4$.

Sets of such subroutines developed in the note books of the programmers and it became common practice to say, "Joe, can I borrow your sine routine?" But it was wartime, and there was never enough time to organize and systematize this information. Soon, however, the EDSAC crew was gathered together, before the computer was finished. They had time to develop and systematize a set of subroutines and provide the first step in automatic coding. They gave us also the first publication in this field, the book by Wheeler, Wilkes, and Gill, "Preparation of Programs for an Electronic Digital Computer." This provided a strong thrust in the direction of building routines to assist the coder.

As new computing systems appeared in which the input-output equipment was under control of the central computer, and in which there were multiple input-output units so that they could be used as secondary storage, a new situation was presented. The first of these data-processing systems was the "UNIVAC® I" Computer and it was a natural development that the first "compiler", AØ (1952), was designed for the UNIVAC I Computer. It was followed by the A-1 and A-2 Compilers. The pseudo-code of the A-2 Compiler (1953) took the form:

AMO 008 000 018	calculate cx
TSO 018 000 020	calculate sin cx
AMO 016 020 024	calculate e^{-x^2}. sin cx

where	c is stored in working storage location	008
	x " " " " " "	000
	cx " " " " " "	018
	sin cx " " " " " "	020
	e^{-x^2} " " " " " "	016
and	e^{-x^2} sin cx " " " " " "	024

Thus the compiler (a routine on tape) plus its library of subroutines (on tape in relative coding) effectively converted the UNIVAC I Computer from a single-address, fixed-decimal computer into a three-address, floating-decimal computer. Use of the pseudo-code reduced the amount of coding to be written by a factor of twenty and with this the number of mistakes by an even larger factor. Further, it was possible to teach this pseudo-code to mathematicians, engineers, statisticians, and other scientists in a few days time in contrast to the weeks required to learn computer code.

This represented a major step forward. However, after a few months in use, it was discovered that the library included about 28 mathematical

subroutines, 37 input routines, and 48 output routines, and that each new problem was tending to require a new input routine and a new output routine. The input routines were examined and it was discovered that they were all of essentially the same nature and based on four parameters;

1. the tape (or servo number) to be read (T)

2. the input storage area through which the data were to be transferred (A or B, or C...),

3. the number of words to be transferred in each item or record (S),

4. the position in working storage to which the quantities were to be transferred (W).

A new routine, called GMI, "generator-move-in", was prepared and placed in the library which was capable, upon receiving the pseudo-coded word as input, of creating or <u>generating</u> as its input the required subroutines. The pseudo-code word was written in the form:

```
GMI OTA OSS WWW
        B
```

or

GMI 02B 010 122 to generate a subroutine in relative coding to read from servo 2, through input area B, in 10-word items, to working-storage locations 122-131. All of the input subroutines in the library were replaced by this generator. A similar generator GM<u>O</u> (generator-move-out) replaced the output subroutines.

These two were the first operating generators. Since that time many have been developed and it is their contribution that has made possible the development of data-processing compilers. The A-2 Compiler performed other services in the course of constructing a program besides calling for static or generated subroutines and processing them. It also allocated the storage for data and instructions, arranged for segmenting if more than a single storage load of instructions was required, and prepared a report on the program it produced.

<u>Algebraic Translation Routines</u>

In 1953, Laning and Zierler, at MIT, set to work to produce a pseudocode more nearly resembling normal mathematical statements than did the pseudo-code employed by either of the major MIT routines. The produced "A Program for Translation of Mathematical Equations for Whirlwind I." This routine was followed in 1956 by the MathMatic Compiler for UNIVAC I and II computers, Fortran in 1957 for the IBM 704, and later UNICODE for the UNIVAC 1103A Computer. All three of these routines accept mathematical symbology and English verbs.

Consider a very small example:

Compute $Y = \dfrac{X^3(2+X)}{3 \text{ COS } A} - P^{\frac{1}{2}}$

for $0.2 \leq P \leq 0.8, \Delta P = 0.2$

$0.35 \leq A \leq 1.05, \Delta A = 0.175$

$1.8 \leq X \leq 3.8, \Delta X = 0.5$

Coded in MathMatic, the problem becomes

(1) VARY P 0.2 (0.2) 0.8 SENTENCES 2 THRU 5.

(2) VARY A 0.35 (0.175) 1.05 SENTENCES 3 THRU 5.

(3) VARY X 1.8 (0.5) 3.8 SENTENCES 4 THRU 5.

(4) $Y = X^3 * (2 + X) / (3 * \underline{\text{CO}}\text{S } A) - P^{\frac{1}{2}}$.

(5) WRITE AND EDIT Y, X, A, P.

(6) STOP.

A meeting was held in Zurich during the summer of 1958 between representatives of the Association for Computing Machinery and GAMM, the German Association of Applied Mathematics and Mechanics, to start to define an international mathematical language ALGOL. Reports of their continuing effort have appeared in the "Communications for the ACM."

A Data Processing Compiler

Meanwhile, with the data-processing systems coming off the production lines, a need became evident for a new type of compiling routine -- one to prepare data-processing programs as contrasted to the mathematical type of programs. Work was begun in 1954 on such a compiler for the UNIVAC I Computer and several serious problems immediately became apparent.

1. There existed in commercial and business systems no common symbology, and, in fact, worse, no common terminology. Not only did no two installations use the same terminology but even divisions within a company differed.

2. No longer was the computer to deal with quantities -- decimal or binary numbers -- fixed in length and type of decimal notation and location. It must deal with alpha,

alpha-numeric, and numeric quantities, in coded-decimal notations, with and without decimal points in various positions. Several quantities were often packed together in a computer word, or a single quantity or field might spread out over several words.

3. The operations to be performed often required many operands, yielded many results, and operands could also become results by manipulation.

These problems were reduced to

1. A language or symbology must be developed, acceptable to all installations by means of which operations could be stated.

2. A method must be devised for describing data.

3. A multi-address pseudo-code must be developed.

While the solutions to these problems to be here presented will seem simple and almost obvious, it must be stated that these solutions were not arrived at directly but via many pitfalls, detours, and over many obstacles. The FlowMatic* 1/ compiling system first operated in 1956.

Flowmatic -- Pseudo-Code

It was determined that the only common symbology used by the various installations was the English language and it was therefore decided to use a selected English vocabulary as a pseudo-code. It was possible to isolate and define 30 verbs.

ADD	INSERT	SELECT
CLOSE-OUT	JUMP	SELECT-LEAST
COMPARE	MOVE	SET
COUNT	MULTIPLY	STOP
DIVIDE	NUMERIC-TEST	SUBTRACT
EXECUTE	OVERLAY	SWITCH
FILL	PRINT-OUT	TEST

1/ *Trademark of Sperry Rand Corp.

HALT	READ-ITEM	TRANSFER
IGNORE	REPLACE	TYPE-IN
INPUT	REWIND	WRITE-ITEM

which, as a start, would make possible definition of data-processing problems in a common manner. Thus, a sentence could be written

(19) COMPARE INV-ON-HAND WITH INV-LIMIT ; IF GREATER GO TO OPERATION 29 ; IF EQUAL GO TO OPERATION 23 ; IF LESS GO TO OPERATION 23.

Even such a simple statement turned out to be fraught with traps! The first user dropped the last two clauses and wrote "OTHERWISE GO TO OPERATION 23". The computer stopped, indicating that it did not recognize the word "otherwise." It was necessary to provide for a large number of ways of ending the sentence including "IF UNEQUAL GO TO OPERATION N." The second user did not choose to compare one field WITH another. He elected to compare one field AGAINST another. It was necessary to inform the compiler that any one of a set of words, WITH, TO, FOR, AGAINST, or even a colon, was acceptable between the two noun-names of the fields. Coping with variations in English statements did not end the problems: next came misspelling.

Suppose someone wrote COMPAIR instead of COMPARE. The first time this occurs the computer stops, states that it cannot locate this verb, and requests a correction. The second time someone makes this same error, the computer stops and types out,

"YOU WROTE COMPAIR, DID YOU MEAN COMPARE, IF SO, HIT THE START BAR"

Thus, the symbology selected for the data-processing compiler became a restricted form of simple English sentences and a preliminary set of verbs was defined. This provided an answer to the first and last problems and left the nouns to be treated.

FlowMatic -- Data Descriptions

While 30 verbs common to all data-processing problems could be distinguished, no such list of nouns could be prepared. It became clear that each installation must be permitted to define its own nouns, thus describing its own data. It was determined that these definitions must include descriptions of

1. The data as a file or set of reels of tape

2. the data as items or records of sets of words, and

3. the data as fields or sets of characters.

In order to describe the data as a file, the noun-name by which the file will be referred to must be given, the label carried on each tape of the file and its position on each reel, and the sentinels used to indicate end of reel and end of file and their relative positions. This information provides the necessary parameters for the input-output generator to supply all of the tape-handling coding including label and sentinel checks, reading tape, and servo-swapping.

To describe an item in a file, in order to position the items for processing, it is necessary that the size be stated (if fixed length) or a description of headers and trailers or an ending convention (if variable length). The names of the fields, or "keys," dictating the sequencing of the file are required for matching, sorting, and merging.

Finally, each field is given a noun-name and descriptor stating the type of field, its relative position in the item, its length, and the position of its decimal point and sign, if any. These definitions and descriptors are stored in the library of the particular installation and are thus available for use in the pseudo-code.

A file might be named "inventory," its label could be INV063000001. It could consist of 10-word items corresponding to each "stock-number" and, if arranged in ascending order of "stock-number," that field would become the key. A field such as the stock-number could be described as positioned in the first word of the item, alpha-numeric, starting in the first digit of the word, consisting of ten digits and having no decimal point or sign. Similarly, INV-LIMIT (inventory limit) might lie in the sixth word, be numeric, start in the seventh digit, consist of four digits, the decimal point positioned two digits to the right of the beginning of the field and no sign. Simple forms were designed to facilitate the writing of these data descriptions. Only a few weeks later a request was received for "a routine to fill out the forms." This routine now exists.

FlowMatic Generators

As may be seen from the data descriptions the operation "compare inv-on-hand with inv-limit" might indicate a simple comparison of two computer words. More likely, the phrase would require that the two quantities be extracted from surrounding fields and shifted to line up decimal points before the comparison could be made. Corresponding to each permissible verb, there is a generator in the library which, upon receiving as input data descriptions and parameters, is capable of generating a relatively coded subroutine to perform the particular operation upon the specific fields named. Obviously, these generators must be quite sophisticated and some, such as the input-output generator, are

extremely powerful. The latter may deliver as a result of a single input sentence between 300 and 2000 coded computer instructions. It is essentially a small compiler including generators within itself. Other generators such as JUMP, MOVE, and EXECUTE proved to be comparatively simple. Three generators HALT, IGNORE, and STOP are almost trivial but turned out to be quite essential.

The FlowMatic System

The input to the Flowmatic system consists of a library of generators (corresponding to verbs) and data descriptions (corresponding to nouns). A problem is written in English sentences -- FlowMatic pseudo-code. Processing by the FlowMatic compiler produces a computer-coded program ready to be run on the Univac I or II Computer (depending upon the library employed). The compiler itself operates in three phases controlled by three routines called Translator, Selector, and Converter.

The Translator processes the pseudo-code and refers to the data descriptions. It produces an expanded problem description. Each sentence of the English pseudo-code is expanded into a "file entry" in the form

OP NUMBER 19	
CALL WORD	
COMPARE	(verb)
VARIABLES	
INV-ON-HAND	(noun and abbreviated descriptor)
0053///16	
INV-LIMIT	
0063///76	
PARAMETER	
4	(sentence ended "if greater" - "otherwise")
JUMPS	
J29	
J23	
END OF ENTRY	

This file of expanded sentences can be considered to be a description of the problem in a computer-oriented language -- if a systematic method of conveying information may be called a language. This "language" is curiously general. It is not based on the word structure of any computer nor are its symbols dependent upon position. Each entry is preceded by an English word indicating the character of the information to follow. It uses the 26 letters of the alphabet, the 10 decimal digits, and characters for "space" and "ignore."

The Selector accepts as input the expanded problem description and controls and provides services to the library of generators. As the Selector scans the file entries it summons the appropriate generators and supplies to them the information contained in the file descriptions as required. Each generator produces the subroutine, in relative code, required by the data descriptions. The generator also sends to the Selector information concerning the subroutine it has produced. The Selector adds this information to the file entry to produce a new file entry which will now contain, in addition, the start line, the end line, the entrance and exit lines, a list of needed constants, temporary storage and working storage, the transfer registers and other subroutines required by the newly generated section of coding.

This second file contains listings of all the components of the final program; storage requirements for data, variables, constants, working and temporary storage, and instructions (subroutines). In addition, it contains the necessary information concerning relationships such as jumps and return jumps, entrances and exits.

The final phase of the compiler, the Converter, allocates the storage in the most economical fashion, assigning specific locations to all data and instructions and finally acting as an X-1 Assembly System, produces an object program. Since at the time of allocation complete information was available concerning all data and instructions, it also produces a report called the "record." The record supplies a copy of the original pseudo-code and the data descriptions drawn from the library. It lists the storage assignments for input and output data and for variables. Corresponding to each pseudo-coded sentence it lists the relative X-1 coding produced by the generator together with the corresponding final fixed coding and its location in the object program.

As the reports came in from the installations trying out the compiler, it became evident that certain things had been proven:

1. the compiler reduced elapsed time of problem preparation

    a) by reducing analysis time a little bit by supplying a vocabulary and a technique for process charting,

    b) by reducing programming time by a factor of at least five,

c) by eliminating coding time,

d) by almost eliminating debugging time.

2. the use of the compiler reduced training time;

3. the compiler assisted in standardizing the formats of data and programs;

4. a new communication system was introduced extending from manager, through analyst, programmer, coder, and operator to the computer;

5. runs could be easily altered and updated by correcting the pseudo-code and recompiling;

6. a permanent record was maintained of the detailed content of each running program;

7. and finally, all of these contributed to a reduction in the cost of programming and operating a computer.

Such economies immediately posed the question "could the system be extended to other computers?"

The Aimaco System

It is not immediately evident, but the first two phases (Translator and Selector) of the FlowMatic compiler are independent of the computer to be programmed. The Air Materiel Command needed English language compilers for all of their computers. The next series of computers to be delivered were the Univac 1105 computers. It was decided to set up a joint project of AMC and Sperry Rand to produce a preliminary Air Materiel Command -- AIMACO -- Compiler making use of all existing routines as far as possible. These included the FlowMatic Compiler (and AMC had a UNIVAC I computer in Dayton) and the USE Compiler (Univac Scientific Exchange) as developed by Ramo-Wooldridge. The AIMACO compiler is then, a hybrid, running on two computers. Three components had to be prepared, a converter, an input-output generator, and a set of generators. The converter was prepared by using the FlowMatic Compiler; i.e., the converter was treated as the data-processing routine that it is, and was written in English sentences. The input-output generator could be prepared using the USE compiler. Finally, the FlowMatic compiler was modified into a new compiler -- G-Ø -- used by the Air Materiel Command programmers specifically for the production of generators. Thus, the components necessary to complete the AIMACO compiler were developed automatically.

It must be noted that this is not a recursive operation. Inefficiencies present in the original FlowMatic Compiler and generators are not

transmitted, they act only as processing agents on the new efficient English definitions. When G-Ø is used to produce a new generator, the efficiency of the new new generator depends on the English sentences defining it -- not on G-Ø. Compilers may be used successfully to re-code themselves to greater efficiency as the English statements are altered and improved.

Extension of FlowMatic Compiler

One means of extending the usefulness of the FlowMatic Compiler consists in designing broader and more powerful generators. COMPUT-FICA is such a generator. It covers about twelve elementary FlowMatic sentences in one comprehensive sentence:

(19) COMPUTE-FICA ON GROSS-PAY, DEDUCT FROM ADJUSTED-PAY AND STORE AS WEEKLY-FICA; ACCUMULATE ANNUAL-FICA, QUARTERLY FICA AND MONTHLY FICA; THEN GO TO OPERATION 27.

Not too many such compound generators can be distinguished for computer applications as a whole, but more can be defined within an installation or a group of installations in a single company or activity.

A Common Language for Computers?

About this time IBM announced the Commercial Translator and several other computer manufacturers announced they intended to make English language compilers. In May 1959 the Department of Defense called a meeting in Washington. The first day manufacturers met together with representatives of the defense installations. The second day the major industrial and commercial computer users joined the discussion. Three questions were asked: Should we have a common data-processing language? Is such a language feasible? Is it time to do something about a common business-oriented language? The agreement was unanimous that a common data-processing language was needed; it was agreed that it was feasible as demonstrated by FlowMatic and AIMACO; and it was agreed further that the time was ripe to go ahead with the definition of a common language before all the manufacturers independently developed different languages.

This development Mr. Charles A. Phillips of the Office of the Assistant Secretary of Defense reported at the meeting of the ACM in September, 1959. The Department of Defense has continued to sponsor this effort. It is important to everyone concerned with computers. The first level committee has reported and offered a language called COBOL -- COmmon Business Oriented Language -- a language which, it is hoped, will be implemented by all of the computer manufacturers. This "first approximation" language is independent of any particular computer and is problem-oriented.

It can best be presented by quoting a part of the introduction to the COBOL Manual (Report to Conference on Data Systems Languages including Initial Specifications for a Common Business Oriented Language (COBOL) for Programming Electronic Digital Computers, Department of Defense, April 1960, U. S. Government Printing Office, Washington 25, D. C.)

To the Conference on Data Systems Languages

Subject: COBOL - Preliminary Specifications for a COmmon Business Oriented Language

At a meeting January 7-8, 1960, the Executive Committee accepted and approved for publication and distribution to the Conference the subject report of the Short Range Committee dated December 17, 1959. The Executive Committee believes that such preliminary specifications for COBOL are a major contribution in the development of a single business data processing language. COBOL represents the only method of expressing business data processing problems acceptable by such a wide group of data processing systems. Most of the manufacturers of data processing equipment have recognized the benefits to all users and to manufacturers of using a common programming language and most of the manufacturers have agreed to provide COBOL compilers as part of their programming service to customers.

In addition to editing the report (and preliminary specifications) for typographical and other minor errors, the Executive Committee rewrote Part I, Introduction. Section 4, Phasing, and Section 5, Maintenance, now reflect the Executive Committee's wish to emphasize the fact that deficiencies in the preliminary specifications are well recognized together with the establishment of a mechanism by which such deficiencies can be overcome promptly and effectively.

The Conference on Data Systems Languages is a voluntary cooperative effort of users of data processing systems (both in the government and industry) and manufacturers of data processing systems. The objective of this effort is to develop a common language, basically in English, which is oriented toward business data processing problems, open-ended and independent of any make or model of data processing equipment. The specifications for such a COmmon Business Oriented Language (COBOL) as set forth herein represent the first milestone toward this objective.

The Executive Committee recommends that all users of general purpose computers consider the use of COBOL in programming business data processing problems.

C. A. Phillips, Chairman
Executive Committee
Conference on Data Systems Languages

The B-2 FlowMatic Compiler which will accept COBOL as an input language, use the UNIVAC II Computer as source computer, and produce programs for the UNIVAC II Computer as object computer, will be completed on 31 October 1960. Compiler development for future computers will continue as the development of COBOL continues.

The B-2 Compiler is but one of the compilers in the FlowMatic System for the UNIVAC computers. It is similar in structure but not identical with the original FlowMatic Compiler--B-Ø. It is presently written in FlowMatic and shortly will be translated into COBOL. The use of the FlowMatic B-Ø Compiler to build the FlowMatic B-2 COBOL Compiler marks another milestone. It is the first time a compiler has been used to make a compiler, thus finally demonstrating that coding is data, that a data-processing compiler can produce programs to process coding, and that a data-processing compiler can be used to produce a data-processing compiler.

Credit for the success of this effort is due to three groups of people:

to the members of the Research Automatic Programming Group[1/] for their courage, initiative, imagination, and hard work in producing these systems;

to our superiors in the Remington Rand UNIVAC Division who have believed in the work and supported the efforts;

and finally and most importantly to the UNIVAC installations, particularly United States Steel, the Hydro-Electric Power Commission of Ontario, the United States Navy, and the United States Air Force who used the new systems and who provided the encouragement to continue toward better systems.

1/ The organizations participating in the original development of the COBOL System in 1959 were:

Air Materiel Command, U. S. Air Force
Bureau of Standards, Department of Commerce
Datamatic Division, Minneapolis-Honeywell Corporation
David Taylor Model Basin, Bureau of Ships, U. S. Navy
Electro Data Division, Burroughs Corporation
International Business Machines Corporation
Radio Corporation of America
Remington-Rand Division of Sperry Rand Corporation
Sylvania Electric Products, Inc.

THE MATTER OF PROGRAMMING

William Orchard-Hays

William Orchard-Hays is Vice President and Director of the Information Processing Technology Division of CEIR, Inc. He is a specialist in scientific and large-scale systems programming, organization of computing machine procedures, and development of computational techniques for Linear Programming and related models. Prior to coming to CEIR he had extensive experience in numerical analysis with the Rand Corporation. He is author of an impressive list of technical articles.

The matter of programming for digital computers can be approached from many viewpoints:

(1) As a profession -- background of personnel, preparation, responsibilities, standing in technical society, salaries, paths of opportunity.
(2) As a technical skill -- detailed training in machine logic and operation with examples, tests, class projects.
(3) As a major cost item -- size of staff, cost of preparing machine instructions, lead times, related overhead costs, etc.
(4) As an administrative and planning problem.

The last will be the main approach taken this morning but we will, of necessity, touch on the others as well.

First, let us be sure we all understand why there is a problem worth spending the rest of the morning on. To those unfamiliar with this field, it should be explained that digital computer hardware systems only provide a sort of universe within which to work; in and of themselves they can handle no problems at all. The processes carried out, the monitoring of these processes, the controls provided, and, in general, the whole application of a computer to real problems is entirely dependent on the program or system of programs provided for the machine, within the limits inherent in the hardware, of course. The preparation of these programs, i.e. computer programming, is a long and tedious process and usually involves the understanding and use of several "languages" -- machine code, symbolic code, more abstract computer languages, and the jargon associated with the particular problem at hand.

To give a measure of the volume of detail involved, one specialized system of routines recently prepared at C-E-I-R for use by major oil companies involves some 13,000 instructions. It is capable of handling any problem in one particular class. Systems with over 50,000 instructions exist.

Preparation and Use of Computer Routines

It has been estimated that, by the time a computer routine is checked out and ready for productive work, each instruction has cost some $5.00. Actually, no single figure holds over a representative sample but it is clear that programming is expensive. The preparation of a routine involves several steps, usually about as follows, with variations due to specific circumstances or individual work habits.

(1) Analysis of the problem or orientation by an analyst.

(2) Evaluation of different mathematical or logical techniques, different logical procedures within the computer, and available canned routines pertinent to the problem.

(3) Detailed flow-charting of the logical and arithmetic processes to be carried out and, usually, allocation of computer storage "cells" and auxiliary units.

(4) Writing of code in a suitable "language" and transcribing it to a suitable medium, e.g. punched cards.

(5) Assembly or compilation of the hand-written code to translate, organize, and record it into the form of machine code which can actually be recognized and executed by the hardware.

(6) Testing and "debugging" of the routine with synthetic or hand-computed test cases.

(7) Final cleanup of the code, documentation, operating instructions, etc.

The assembly or compilation of step (5) is done by the machine itself by means of a pre-existing routine. These latter are usually very elaborate, and enormous amounts of effort have been devoted to devising good "assemblers" and "compilers." They are analogous to the machines which the machine tool industry uses to produce machine tools. Steps (3), (4), (6), (7) and in part (2) are built around the "language," philosophy, and conventions of the assembler or compiler used in (5). As a measure of the effort involved, development of IBM's FORTRAN Compiler has by now consumed well over 30 man-years and untold machine hours.

In using a checked out routine for productive work, there is also a series of steps of which the following are fairly typical:

(1) Collection, editing, and transcribing of data and parameters;

(2) Loading of the routine and data into the computer;

(3) Processing of the data by the computer;

(4) Transcribing of output;

(5) Checking and evaluation of results.

Here again, steps (2), (3) and (4) are carried out by the computer, probably with peripheral equipment being involved in (1) and (4). Since steps (5) and (6) of the routine preparation process involve this last process, with the assembler here the "productive" routine, and since both kinds of procedures must be carried out in great variety throughout the day at any one computer installation, there is often a severe planning, scheduling, and operational problem. Futhermore, the debugging process is complicated and requires communication with

the assembly and output functions to be effective. This whole complex can be further complicated by the development and use of elaborate, specialized systems which engender many difficulties related to the problem, the data, and training of the user, quite apart from the programming or machine "bugs."

More General Considerations

Now I think you will agree that we could stop right here and have enough problems to chew on for some time. Actually, however, this is only the beginning. It has been implicitly assumed that computer applications can be broken down into independent, manageable, "bite" sizes which can be routinely specified and processed through a sort of job shop. If this were true, there would not be much more to say except to recommend that your technical people study and attend classes on existing operating and compiler systems since extensive work has already been done on the procedural problems just outlined. Indeed, it may be true for some of you, depending on the applications contemplated. But more generally you will find a growing need for integrating many stages of computing or data processing into one grand operating procedure. In fact you may find the need for maintaining several such grand procedures plus provisions for job shop work, simultaneously. It is at this point that strong policy guidance is essential if the whole operation is not to become mired in confusion and inefficiency. This guidance can be a delicate matter since administrators and technical experts may clash head-on unless both exert great effort to grasp the other's problems and, more importantly, the larger common goal which both are seeking. Sounds sensible, doesn't it? But, as no doubt you're aware, it isn't easy.

In regard to overall procedures and systems, one idea is of prime importance -- the planning for the large projects must strike a realistic balance between the technical sophistication appropriate to the machine and the problem on the one hand, and the practical requirements of the project in the content of other installation commitments, on the other. In other words, give the client a professional job but don't get carried away with sub-optimization of only one part of your overall responsibility. No doubt, as managers, you do not need to be told this in any ordinary managerial content. But ADP is not ordinary; it has a compelling and almost seductive fascination, much like an expensive electric train has to a mechanically inclined boy, or his father. Perhaps an example is in order.

Setting: Computing shop engaged primarily in engineering and modest scientific calculations. However, certain percentage of machine justified by personnel and payroll departments to process payroll, cost distribution, and maintain personnel records file.

Administration: Autonomous department reporting directly at high level with responsibility to provide required services, meet payroll-personnel commitments, promote developments in computing (or equivalent), and with freedom to select methods and procedures.

Technical Planning: Chief programmer-analyst, with long experience in job shop type of operation with engineering computing, promotes and gets authority to install a comprehensive operating system of programs

which makes possible batched processing of heterogeneous small jobs with automatic machine time accounting, standardized operating and malfunction procedures which are independent of the current job, a large library of standard routines, etc. In the name of uniformity and saving machine set-up time, the payroll and personnel files are programmed to run in the same systems.

Results: The engineering department is delighted as are the contract administration people. The computing department and customer relations people make a great to-do over their efficient, smooth running machine room. Automation is great. But the payroll people are always in a bother because an error in their data causes so much repercussion in the operating system that it always takes until 4 A.M. to get a good check-writing run and they spend the next 4 days hand-correcting their records. Personnel department can't understand why they don't get better service on information requests and why the machine time they get charged with is five times the estimates on which the original justification was based. The O.R. group, after a few sad experiences, restrict their use of the machine to a few standardized statistical calculations. As to the head of the computing department, he is relieved of any need to promote improvements in computer technology since his continually increasing machine utilization is prima facie proof of the importance of his department.

It can be left to you to decide whether this mythical department head is discharging all his responsibilities equally well or whether the above fable has application to your own situation. It did seem obligatory to bring to your attention the possibility that automated systems can so mitigate the ability to experiment and to make decisions that man becomes a slave to his own inventions. After all, systems exist to serve some human purpose, not to be self-sufficient automatons. The more sophisticated the mechanism, the more it should be considered a member of a man-machine team. The trick for making hardware a useful member of a team is to construct systems which are self-adaptive in a machine-oriented sense, but externally responsive in a problem-oriented sense. The know-how to accomplish this is evolving and already exists in some fields.

Languages and Processors

There have been two developments in programming during recent years which deserve our attention. First, there have been deliberate efforts to create automatic, comprehensive operating systems for computers. These are aimed at organizing and simplifying the jumble of operations previously described into something that can be easily scheduled, operated, and accounted for, with a minimum of human operators. At the same time, extensive provisions for the programmer's needs, both in writing and debugging a routine, are included. Undoubtedly the most ambitious of these projects has been the so-called SOS (SHARE Operating System) which was designed by a committee of a cooperative user's group for the IBM 709 and is immediately transferrable to the transistorized IBM 7090. The actual programming of the system was undertaken by IBM with assistance from user organizations. The project is now some $3\frac{1}{2}$ years old and the system is still imperfect, though workable. There is currently

considerable difference of opinion as to the value and general utility of such a system. It is the author's opinion that the main disadvantages of it are that, while exercising a high degree of control over the machine and its application to problems, it is neither adaptive to varying conditions nor responsive to the user's peculiar needs. This project has certainly been a trail blazer in demonstrating new concepts, techniques, and degrees of systemization and it will no doubt be studied and "dissected" for parts for a long time. However, it seems to be an example of automation gone rampant.

The other, and somewhat older, development in computing technology has been the creation of so-called automatic coding systems. The intent here is to devise languages easily learned by technical people and to write "processors" which convert these more abstract languages to machine code automatically. The processors are themselves elaborate systems of programs which have to be hand-coded for each type of computer. In order to make both the language and the processor foolproof, restrictions have to be imposed which, for some problems, severely curtail the facilities of the machine or, what is equally bad, add to already ponderous programming and operating practices. Most "languages" to date have been algebraic, though business data processing languages are in the works, two or three elementary ones being already available. But the same charge can be levelled against the automatic coding systems as against the operating systems: they demand too much with too little responsiveness to the user's peculiar needs. Again, however, they have demonstrated the value of certain basic concepts, in particular, the power of a system which permits the user to communicate with the machine in a language closely related to his problem. Since many problems can be stated algebraically, the algebraic compilers have achieved considerable success. From a broader point of view, this is only an accident. It will be interesting to see how wide a utilization any particular data processing language will enjoy lacking a background comparable to mathematics.

Example:

To illustrate the differences in the "languages" previously referred to, the FORTRAN language for the IBM 704 provides a good example. Let us start with actual machine instructions, which are stored in high speed core storage in the same form as numbers or other data. The 704 has a fixed word-length, that is, its core storage is arranged to have a certain number (4096, 8192, 16384 or 32,768) of "cells" each of which contains a fixed number of binary digits, or bits, since the 704 is a binary machine. Most computers have this fixed word length structure as opposed to a variable length in which the storage is considered as a number of characters, each addressable. (The IBM 705 is such a machine.) In the fixed word length set-up, only complete words are addressable and all the arithmetic and control registers are designed to handle the standard word, or whatever part thereof is appropriate.

The 704 has a word length of 36 bits which, for recording numbers, is considered as 1 sign bit and 35 bits of precision, equivalent to a little more than 10 decimal digits. For recording instructions, the 36 bits are considered to be divided into 4 groups: 3 bits for a "prefix"
15 bits for a "decrement"
3 bits for a "tag"
15 bits for an "address"

The actual operation is recorded in the prefix and decrement. (The two have distinct functions in a small class of operations.) The address refers to a storage cell which contains the operand for the operation and the tag modifies the address for indexing purposes, e.g. for selecting one element of an array stored in consecutive locations. The 704 is a single address machine, as opposed to multiple address machines. For example, if A, B, C are the addresses (i.e. the numerical designations or names) of three storage locations and it were desired to form the sum of the contents of A and B and put the result in C, the 704 would require three instructions, substantially as follows:

(1) "Take" the contents of A,

(2) Add to it the contents of B,

(3) Store the result in C.

A 3-address machine, on the other hand, would require only one instruction which would be substantially,

Add contents of A to contents of B and store in C.

In the single address machine, the programmer has to keep in mind not only the storage locations but also the "current" contents of a register in which the sum (or difference, product, quotient, etc.) is being formed. This is called the accumulator and its name is an implicit second address in many operations. Thus the "take" operation in (1) above implies that the contents of A will be moved to (i.e. reproduced in) the accumulator.

The actual descriptive names of the 704 operations in (1), (2), (3) above are:

Clear and add	(clear accumulator and add from storage)
Add	(add to accumulator from storage)
Store	(from accumulator to storage.)

The actual bit configurations for representing these operations would be very awkward to write in the binary number system since an entire instruction would require 36 marks, i.e. zeroes or ones. To reduce this, actual machine code is written in octal notation since 3 bits can be mentally grouped for an octal digit and the 12 octal digits thus required are both shorter to write and easier to read. Furthermore, the four parts of an instruction would fit this octal representation giving

1 octal digit for prefix
5 octal digits for decrement
1 octal digit for tag
5 octal digits for address

Thus, for practical purposes, 12 octal digits grouped as above on a line are considered to be the "pure" machine language representive of an instruction. If we assume that A, B, C above refer to octal addresses 1000, 1400, 2000, respectively, then instructions (1), (2), (3) appear in machine language as follows:

```
(1') 0 50000 0 01000
(2') 0 40 00 0 01400
(3') 0    00 0 02000
```

Now the 704 has (i.e. recognizes) approximately 90 different operations and it would be difficult to remember all the octal representations together with several other secondary rules and conventions which we have not gone into. (Some people still prefer to code small and medium sized computers in numeric op codes but the practice is becoming passe for several good reasons.) More importantly, however, the assignment of actual numeric addresses at programming time is extremely restrictive. It is like trying to solve a problem in algebra without using X for the unknown, which is almost tantamount to not having algebra as a tool. Consequently, so-called "machine-language" coding for the 704 is actually done in a symbolic language called SAP (Symbolic Assembly Program) which is then converted to actual numeric machine language on the machine by a pre-existing processor called the SAP assembler. There is an official set of mnemonic op codes and the programmer is free to assign symbolic addresses within certain fairly loose restrictions. Thus the programmer would code the three instructions above as

```
(1")        CLA A
(2")        ADD B
(3")        STO C
```

where A, B and C are optional designations. Even though this coding is much more convenient than numeric machine language, there is a one-one correspondence between the two (except for certain additional convenience features of the SAP language which need not concern us here.)

The approach taken in coding in the FORTRAN language (FORmula TRANslation) is quite different. The above "program" would be written simply as

```
C = A + B
```

as though it were an algebraic formula or statement. The meaning of A, B, C here is quite different. They are no longer the names of storage locations but rather the names of variables in a problem. The FORTRAN processor will, of course, assign storage locations for these quantities but the programmer is concerned only with his problem variables. Also, the programmer thinks in terms of algebraic operations rather than machine operations. To illustrate the difference further, the solution of one root of a quadratic is coded in FORTRAN and in SAP. The asterisk is used in FORTRAN to denote multiplication and the slash for division. The quadratic equation is of the form

$$AX^2 + BX + C = D.$$

In FORTRAN, X is found by the statement

```
X = (-B + SQRTF (B*B - 4. *A*C))/2.*A
```

which, except for SQRTF for "square root of" and the stiffness of singly aligned keypunch characters, is remarkably like the usual algebraic notation

$$x = (-b + \sqrt{b^2 - 4\ ac})/2a.$$

The following SAP coding for the same problem only partly illustrates the differences in the two languages. It would be much too complex, for example, to discuss the requirements of input and output here. The following op codes and symbolic addresses will be understandable to a 704 programmer; they are not explained here since the purpose is to illustrate two methods and not to teach SAP coding. It is assumed that a subroutine for taking a square root is available.

```
CLA A
FAD A           FORM 2A
STO TEMP        SAVE
CLA C
FAD C           FORM 2C
LRS 35          PUT IN MQ
FMP TEMP        FORM 4AC
STO COMMON      SAVE
LDQ B
FMP B           FORM BB
FSB COMMON      BB -4AC
TSX SQRT,4      GO TO SQ RT SUB ROUTINE
FSB B           -B + (SQ RT)
FDH TEMP        DIV BY 2A
STO X
```

Now it should not be supposed that the shorter FORTRAN code will result in shorter machine code and faster operation. If anything, the opposite will be true. Also it will take longer to compile the FORTRAN code than to assemble the SAP code. Nevertheless, for many problems, FORTRAN has obvious advantages.

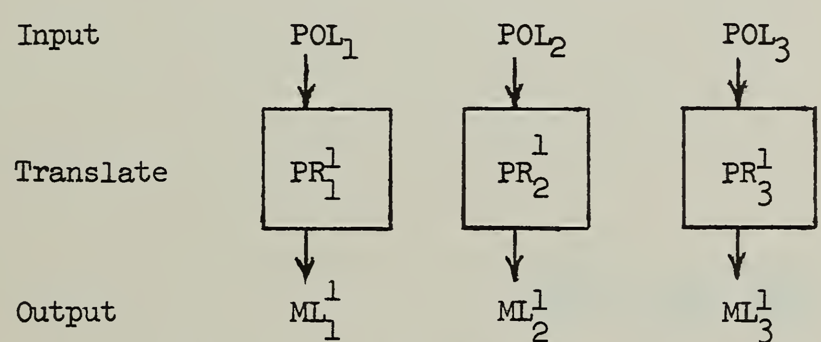
Problem oriented languages for business data processing are quite different still. This is primarily due to the need for defining files of information, the properties described in them, the internal and external formats of the "fields" in which this information is recorded, etc. Also the operations desired are often of a different character. For example, some processors allow the single instruction SORT (referring to a file of data.) Clearly this implies that much detailed information has been supplied somewhere and is available at execution time. There is much less in the way of tradition and custom to guide the designers of such systems. However, a whole new discipline of information handling seems to be evolving. For examples, see the SURGE compiler manual for the 704, the 9PAC and COMTRAN manuals for the IBM 709/7090, the FACT manual for the Honeywell 800, and others.

New Developments

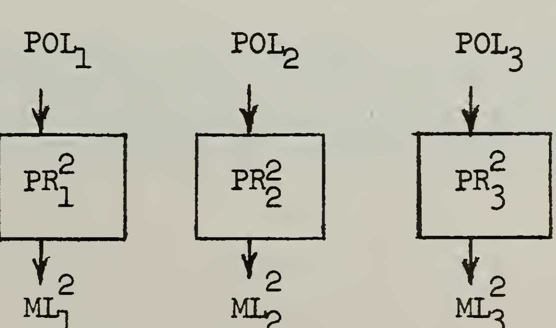
Another interesting line of development in programming is in so-called universal languages. It is much too early to say whether these will ever prove out to the full extent of the claims of their promoters. The problems to which

these people are addressing themselves are very real, however, and deserve some discussion. Universal languages can be considered in two major branches: problem oriented and machine oriented. The former are usually divided further into algebraic and data processing languages. There is an international effort underway to create an algebraic language or family of languages. (See SHARE reports on ALGOL.) For the present, FORTRAN is the nearest thing available (processors exist for several machines) but it is far from universal either in scope or in usage. The Federal Government is sponsoring an effort known as COBOL, for COmmon Business Oriented Language, in the hope of standardizing data processing practices. However, consideration of a universal computer oriented language offers more toward an understanding of the basic problems. Oddly, this appears to have less chance of practical success than the problem oriented language processors but is of more theoretical interest. (See reports of SHARE UNCOL committee. Also, see two articles in January/February 1960 issue of DATAMATION magazine on UNCOL and COBOL.)

To show why a universal computer-oriented language (UNCOL) would be so desirable, consider, say, three problem-oriented languages and their respective processors for one machine. Denote these by POL_1, POL_2, POL_3 and PR^1_1 , PR^1_2 , PR^1_3 , respectively. Denote a machine language code for the machine by ML^1_i Then the use of these processors can be shown graphically as follows:



Now suppose that we wish to be able to process the POL's on another machine. Then it is necessary to prepare three new processors PR^2_1 , PR^2_2, and PR^2_3 to use as follows:



Each of these processors will be expensive to create. Furthermore, if someone devises a useful new language, POL_4, and it is desired to use it on both machines, two new processors, PR^1_4 and PR^2_4 will have to be written. In short, for m POL's and n machines, it is now necessary to have mn hand-written processors.

Now if an UNCOL can be defined, it will only be necessary to have m + n processors, m to translate POL's to UNCOL and n to translate UNCOL to ML's. The former have been dubbed "generators" and the latter "translators". Presumably the m generators could all be programmed for one popular machine, though this is not necessary. The thought is that the n translators would be provided by the machine manufacturers. Certainly this is an intriguing concept but the question remains as to whether or not such an UNCOL can indeed be defined. It is probable that the effort to do so will, in any event, help to clarify some of the most difficult problems in the programming field and help to standardize their statement and investigation.

Another approach to this complex of problems is to develop self-modifying and partially self-organizing central, or "kernel", processors which are heavily machine-oriented but which can be expanded and adapted to particular problem areas. Efforts along these lines are already underway. The emphasis here is more on standardizing the procedures for handling problems and eliminating redundant effort by human beings. The work involved in creating new operating systems will be reduced only in elimination of time consuming details and clerical errors. The success of this effort is heavily dependent on good discipline among its users. Partly because of this, it appears very promising. It may well be that these two approaches will eventually merge as we gain greater insights into the basic problems.

PART 4

ADP SYSTEMS ANALYSIS AND DESIGN

ADP SYSTEMS ANALYSIS AND DESIGN

For Planning, Predicting, and Management Controls

Ezra Glaser

Ezra Glaser is Chief of the Applications Engineering Section of the Data Processing Systems Analysis, National Bureau of Standards. Previously he was on the staff of the Office of Statistical Standards, Bureau of the Budget, where among other things, he contributed importantly to the development of economic and logistic models. He was later a member of an operations research group at National Analysts, Inc., taught a course on systems design with Dr. Alexander at George Washington University in 1960, and is scheduled for a somewhat similar course at the Graduate School of the Department of Agriculture in 1961-62.

At the next session, Mr. Kiefer will review the fairly well-developed procedures that have been shown to be necessary to a competent study of a possible computer application to an information processing task. It is useful to maintain a distinction between these applications and those in which the principal task is the computation of numerical values with the use of a specific mathematical formula, or in which the computer is a part of a physical control process, as in the case of an automatically controlled industrial process. In the ADP studies considered here and further to be described by Mr. Kiefer, the critical operations are the manipulation of information relative to the carrying out of an administrative operation, possibly through the making of routine decisions in most of the cases entering the operating organization. There may also be some not-quite-so-routine decisions to be made. A great deal of importance can attach to this point, so it will be given more detailed attention further along.

The problems to be treated in this session were at least hinted at in the earlier sessions. There was mention of tracing the flow of information through the organizational unit under study. What is it used for? Who makes what kinds of choices, and with the aid of what information? If these questions are to be read as the routine or clerical job of the shop, there is another problem that needs attention: Where does management get its information? The implication here is that management concerns itself with policy, not routine. Perhaps a more difficult question is also going to arise: What information should management get, if there is to be an overhaul of the schemes by which information is recorded, assembled, filed, classified, summarized, tabulated, and printed out for human readers?

A central question in a proper feasibility study deals with a scheme for the way (or the ways) in which the work of an organization can be made to flow through a computer. This session will treat some of the techniques available to construct schemes of this kind. The general term for this form of scheme is a "system," and the art of constructing it is "systems design." A resourceful system designer might have occasion to use many formal techniques in laying out the logical machinery of his proposal. Many familiar tools of quantitative analysis may have application: statistical sampling, the design of experiments, the drawing of logical trees. Some new ones might also be useful: linear programming, inventory control models, game theory, and information theory, among others. Yet it is not obvious that elegant techniques will have a useful place in any particular study; they tend to be applicable only under rather special conditions, and they are heavily dependent upon the possibility of obtaining reliable information about the present operation of the system and the problems with which the organization must contend. In general, it is the techniques that deal with the collection and organization of quantitative data that tend to be useful in a large proportion of the studies leading to the design of a system for the conduct of information and decision routines.

We shall see, then, the application of some familiar older techniques and possibly some newer and less familiar ones in the business of constructing the formal machinery which constitutes the plan for the organization's work to be performed by a computer. These are the tools of the systems designer.

How does one start to design a system in a particular case? It is most efficient to go all the way back to the objectives of the organization and its assigned program or mission. One of the reasons for this is to free the inquiry, for a time, of the present ways of doing things: the familiar practices, conventions, habits, and compromises. They seem inevitable in a world of typed papers, penciled notes, and perhaps punched cards, but they have to be re-examined. As elementary a practice as the filing of names associated with cases under action cannot be taken for granted. Does such a file merely express the present methods, or is it a part of the objective itself? If it is the former, can a system be designed that makes the files unnecessary? If it is necessary, who must have access to it, in what form, why, and how often?

The concern with objectives lies behind some of the advice given in the earlier sessions of this seminar. Typically it is the more experienced and highly placed executives who know basically what the agency is trying to do. Personnel who are placed lower in the management hierarchy are more inclined to be absorbed with procedures and conventions, rather than policies relating to basic missions. Hence the support of highly-placed executives and their active work on ADP studies is usually sought.

What part of the operation can be turned over to a machine — that is, to a strictly formalized system of information processing and decision procedures, presumably to be carried out by a computer system? There are many subtleties in the answer to this question, but it pays to start with a conceptually simple aspect of the study and see where it leads.

Take the case of an agency that processes applications from individuals who seek some benefit to which they claim they are entitled by law. Most of the time it is possible to determine whether the individual qualifies for the benefit by a relatively simple screening battery. There might be alternative paths to the goal. A veteran, for example, might have to prove either a period of adequate length of active service between specified dates, or he might offer a longer period in reserve status, or he might qualify by other actions or by some combination of two or more of these offerings on his application. But this stage might be only a necessary one to obtain the sought benefit, not a sufficient one. Even with these requirements satisfied, it might be necessary also to establish the right class of military discharge, and to make application for the benefit before some expiration date. The various possibilities can be represented by a logical "tree," which can become pretty complicated. (See Figure 1 on page 120 for a simple tree).

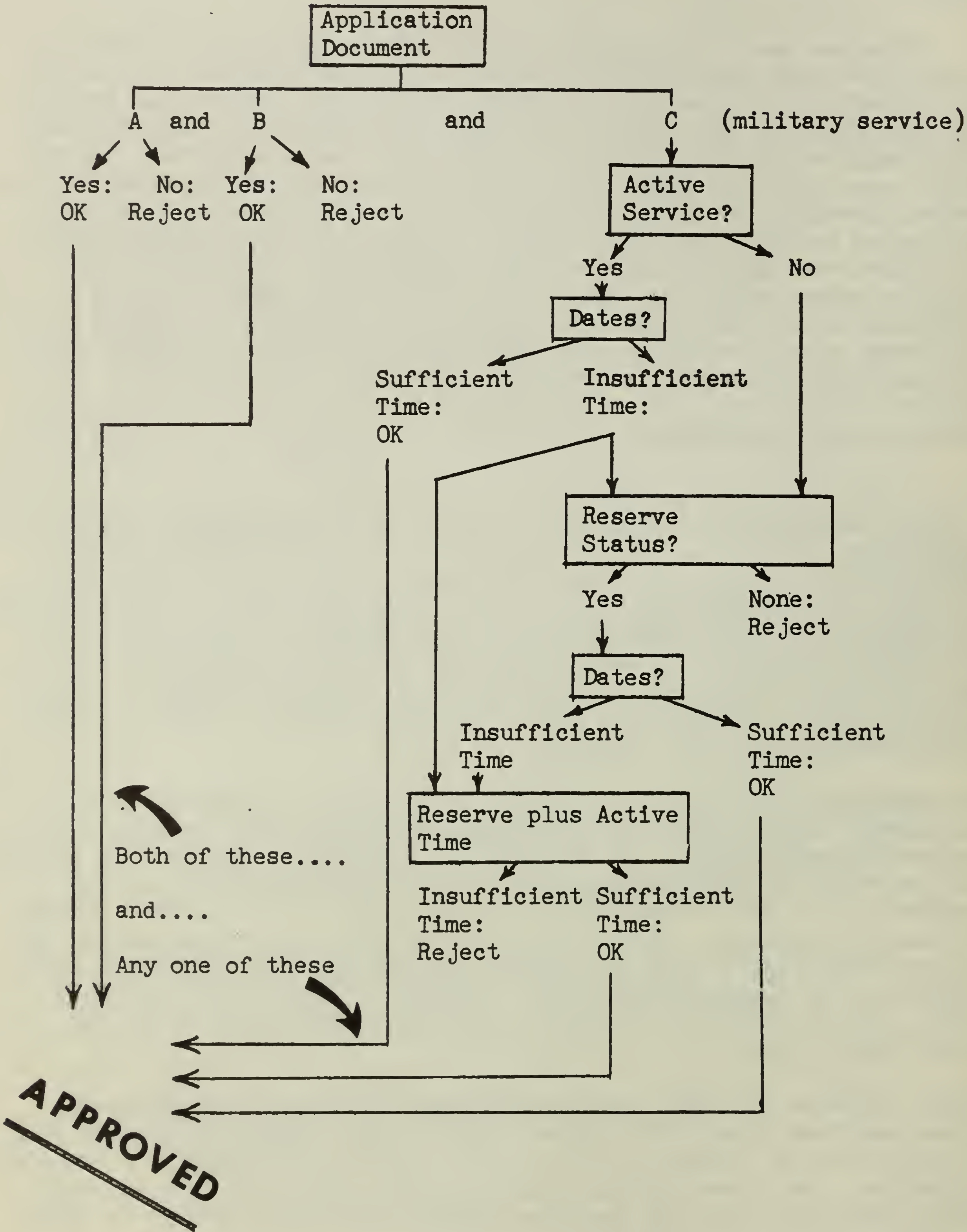
((Note on Figure 1: Each of A, B, and C are necessary but none is sufficient, nor are any pairs sufficient. To qualify or be approved, each case must satisfy all three conditions. The "reject" that follows from the "no" branch of either A or B is sufficient for a complete rejection since both are necessary conditions. However, condition C can be met by active service, reserve status, or a combination of both. As in the above cases, any branch that leads to a "reject" is final. Final approval, then, requires the "yes" branches from both A and B, plus any of the "sufficient time" branches of C.

An important addition to this diagram would be the volume statistics for each path. Hence, if most of the disqualifications were because of a failure to meet condition B, it might be efficient to perform the screening operation on this condition first, thus decreasing the workload imposed on the rest of the logical tree, whether it is machine operated or manually operated. However, policy considerations might dictate the following through of each case even after the question of qualification (approval) or disqualification is resolved. Although this increases costs, it also provides information upon which the appraisal of the rules might be made.

After the details of the information requirements and the operating rules have been added, the logical tree will have been transformed into a systems design suitable to use for the feasibility study of a computer operation. The system design might also have other features and information depending upon the specific context and subject matter to which the logical tree is related.))

Figure 1.--A Logical Tree

Hypothetical Example: To Qualify for a Benefit an Applicant Must Meet Conditions A and B and C.



Conventional procedures experts or organization and methods personnel often make a similar schematic in their "engineering" of a procedure. But their objective is different from the present one in a subtle and important way. The usual job of the procedures man is to break the job down into pieces that can be handled effectively and economically by relatively untrained personnel. The few troublesome cases can be screened out and passed up the line for more experienced and better trained attention. Now the critical question is: can a computer be used to make the same kinds of choices as the modestly trained personnel?

What is the core of the difference? The critical difference is in the complete inability of the computer to exercise even the most modest amount of common sense. The least experienced clerk is given credit for some degree of discrimination in the breakdown of the job into simple units. The computer must have everything reduced to rules. There must always be a definite answer. The logical tree might be long and involved -- even tiresome by human standards -- but for a given collection of initial conditions the answer must be completely specified and it must always be the same for a given input. The earlier sections of this conference, especially those on programming, have made this point. Why, then, does this entire exclusion of elementary judgments make so much difference?

The new level of difficulty which confronts the systems designer comes from both the exclusion of judgment at all mechanized levels of administration and the frequent absence of any real information on the nature of the judgments that are inherent in an existing system.

A straightforward approach would seem to consist of asking everyone who is part of an operation just what he does and what kinds of judgment he exercises at each step. This is not at all a simple question. Moreover, the matter becomes more perplexing if you ask him how frequently he makes each of the kinds of choices he tells you he is required to make. It is likely that he never had occasion before to know how frequently in the course of his work a particular circumstance arose which required each kind of judgment-based choice.

Yet, the design of a computer procedure for doing the job will require that he answer the first, and the economics of the design requires that he also answer the second question. How avoid this dilemma?

The essential clue can often be obtained by rephrasing the question about the kinds of decisions made and their frequency; what do the files (written records) show about the nature of the decisions and their frequency? If the actions are made a matter of record, this approach directs our attention to the contents of the files.

Now we are on more familiar ground. We are concerned with the contents of a possibly voluminous record. For convenience, let us make

some assumptions. Let us suppose that the file reveals all cases where the required action can be specified by a few simple rules.

In one type of situation, the applicant has the necessary citizenship status, service, etc. to meet all requirements for approval.

A second type of situation may involve cases in which the application is complete, all information is present, but some condition required for approval is not met, which of course means rejection.

If an application is incomplete -- some vital piece of information is lacking -- a routine inquiry would be sent to the applicant. Upon receipt of the information, the application would be complete and requisite action can be taken.

Another category is a bit more problematical. Here some information is provided, but it is incomplete or ambiguous. What was done in such cases? Can these actions be codified into rules? In these cases, and in general, will the record clearly reveal the nature of the decisions that were made and the rules implied by the final actions? Let us also assume, finally, that any case which was regarded as atypical or demanding of experienced judgment contains at least manuscript notes indicating what actions were taken and why.

In principle, the behavior of the organization could be pretty well learned by a reading of the entire record of case histories from the first case to the most recent. The nature of the rules being used by the personnel of the organization could then be defined, exceptions noted, precedents discovered. In any practical situation, this is an impossibly burdensome, costly, and time consuming procedure. The problem is solved by <u>sampling</u>.

Sampling offers an additional advantage which might make possible something vital to the study: the close control of the examination of the records of the sampled cases. The sample study allows more careful selection of the limited size staff than would be necessary for a complete reading. Better training and supervision can be provided. Quality checks can be maintained.

But some technique is required. One cannot merely go into the files and "select some cases." The sample drawn must represent the entire file. It cannot be based upon someone's judgment about "typical cases," or it would be better to ask this expert about the contents of the file in the first place. Nor can anyone specify a "cross section" of representative cases without serious danger of biasing the results of the study.

Nor will it do to thrust a card into the file every (say) three inches of file length and select for study the cases into which the card has been inserted. This would yield a sample that is greatly

biased toward the inclusion of thick files. It would probably lead to an erroneous conclusion that most of the cases were elaborate and complicated.

Texts on mathematical statistics deal with the techniques of unbiased sampling. Satisfactory samples can often be drawn from a list of names of cases in the file or of serial numbers or file numbers, provided there is no bias in the assignment of numbers or in the arrangement of the file. If this device can be used, such a "systematic sample" might be well advised. (Incidentally a true probability sample will also yield an effective measure of the sampling variance, so that the investigator will be able to determine the likely boundaries of variability that arise from the decision to sample instead of to read the entire file.)

Note what has been done. A sampling of the file has based the study of the decision rules and the information on which decisions are made upon the contents of the record instead of "expert" opinion about the cases. Moreover, the relative frequencies of all sorts of occurrences among the cases can also be determined from the sample of the record. Note especially that the use of strict principles of mathematical statistics does not arise from a real or imagined need for accuracy to several extra decimal places. It arises from the need to free the study from the preconceived notions of those who are familiar with the system. There is nothing pernicious about this. Anyone who works at length with an administrative routine, for example, has impressions about it, possibly exaggerating the abundance of problem cases simply because these are the ones that drive up the consumption of aspirin.

Now, how to determine what rules are in use? Some of them are in the manual. The only questions that arise are the possible obsolescence of the rule book or the compliance of those who do the daily work. These checks can be made by matching the actions in the sampled cases against the rule book.

But there are also other rules. They are sometimes called "behavioristic rules" to distinguish them from "statutory rules." They represent the customary or habitual way of doing business. A recent example from a regulatory agency showed that there had never been a challenge to an allegation of U. S. citizenship by naturalization in the millions of applications processed. There was an "in fact" rule that directed the clerk or examiner to assume that such allegations were to be considered correct and not to be checked. No such rule had been written anywhere; but the organization behaved as though such a rule had been written down, and honored with full compliance.

This will not do for a computer. The programmer wants to know what to have the computer do for each contingency that might arise. The rule is either to check or not check an allegation of citizenship by naturalization. The computer has no understanding of the usual way of doing business. It has no habits. It has no sense of the appropriate.

Writing down all of the rules that govern a family of administrative decisions can be a most exacting task. In many cases it is easier to study the record and deduce the rules from the "behavior" of the record. Expert help will still be needed for problem cases, but the specification of rules will proceed on the basis of an unbiased sample of actual cases from the history of the organization.

Now we come to a difficult situation: there are some cases for which no definite logical sequence of rules can be specified. No logical tree, even an elaborate one, will suffice. The play of "experience," "background," and "understanding" come into the actions recorded on the record. In some agencies or organizations, a substantial portion of the file will be made up of these cases. Perhaps a U. S. Circuit Court would have nothing in the record of its judicial actions except cases of this kind.

This is possibly the signal for the computer to bow out. If there is necessarily frequent and unavoidable recourse to "experience, background, and understanding" which cannot be codified into rules, no matter how numerous, there might be little a computer can be made to do to take over the operation of the shop.

Even here, however, there are some options. The most obvious one is to stop deciding the problems on a judgmental basis and to substitute a set of rules that produce similar results. Accounting, for example, is full of conventions based upon convenience for the purpose of preventing long adjudications that would not make substantial difference. There is often an option to depart from a convention if one wants to argue some special circumstances. Assuming that some rules can be invented which produce approximately the same results as a judgmental procedure -- and a lot of weight is being carried by the "approximately" -- should the strictly formalized rules then be substituted for the less formal judgment? Obviously there is no general answer to this question. It will depend upon the mission of the organization and the policies and programs that the top executives are trying to effectuate.

Notice that the choice to substitute a rigid set of formal rules is not the business of the computer programmer or of the systems design expert. They might try their hand at the construction of such sets of rules or logical trees. But the decision to formalize is that of the chief officer or the top group of the organization. If it does not make sense to them, the rules cannot be used.

What would motivate the choice to compromise an accepted set of procedures by the introduction of arbitrary elements of rigid formalism into the operation of an organization? The principal objective is to extend the range of uninterrupted computer processing -- data processing -- of the agency's business. Every time judgment is required, the computer must bow out. In practice, this means that the computer system

must print out or otherwise communicate to the outside world that it has reached a situation for which it has not been given rules (other than the rule to refer to a human choice). After judgment has been exercised, a new input is prepared so that the machine processing can be resumed.

The four sessions of this conference which dealt with the nature of the computer equipment emphasized the high cost and loss of time associated with repeated interchanges between the machine world and the human world. If there is any substantial amount of this, it is most likely that the economics of the proposed computer application will dictate against the use of machines altogether.

We have avoided one interesting possibility. It is the case where the computer cannot make judgments but a routine can be devised which will effectively identify those cases which cannot be handled by the assigned rules. These cases might be left entirely to human handling in the form of a "problem printout." If there are sufficiently few of these, they might be removed from the machine system even if a set of rarely used rules might be devised. This is done to simplify the rules (i.e. the computer program) at the cost of a few manually processed cases.

How many cases is a "few"? This depends upon the actual situation. Note, however, the importance of a good estimate of the frequency of occurrence of various kinds of problem cases in the files, which was cited above as the reason for using proper sampling methods instead of intuitively-acceptable or slipshod procedures. The economics of each case must be worked out by the systems designers. What kinds of cases require human judgment? How often do they occur? What savings in time, money, and convenience would attend a given substitution of rules for judgment? How different would be the actions of the organization? After the trade-offs of convenience against formality have been specified by the systems design technicians, the subject matter experts of the agency make their choices. In principal there can be no appeal. There is nothing in the training of a computer technician to place him in a position to disagree with those whose business he seeks to adapt to machine procedures.

For simplicity a case has been overlooked: that where inconsistencies show up which are not wanted. If there is one consequence of a proper formal book of rules, it is that results will always be consistent. A sampling of the file might reveal meaningless or unsought variations in the disposition of identical cases. These can be caught by the review by subject matter experts of the unexplained variations in the files. Or, if no rules can be found for a particular set of variations, the agency might be quite willing to eliminate the variations. People usually expect consistency of treatment from government agencies, and often from private businesses as well, so the mechanical use of batteries of rules is not, in itself, onerous.

The objection will be to the production of unacceptable or even silly outcomes. It is important that the outcome must not be unacceptable or silly to the organization and to the subject matter experts. The computer technicians do not have a vote.

The organization is left with two difficult species of problems that it cannot delegate away:

(1) It must unmake the compromises of a lifetime with regard to the acceptable ways of doing business, and reconsider the acceptability of formal rules in place of well-bred habits of administrative behavior; and

(2) It must make choices between the convenience of formal rules with the consequent modifications of acceptable patterns of decisions, and the inability of computers to carry out routines that require judgment.

The title of this session referred to "management sciences." They are usually considered to include mathematical statistics and the formal design of systems. Another part of the general area of quantitative technique for the use of management deals with the problems of providing the information that management needs to do its job. We are no longer talking about administrative or clerical routines; the concern is with the policy making and overall supervision of an agency.

Again comes the problem of acceptable compromises. The executive can sometimes ask for "all of the information that can readily be made available," knowing that he can scan it for the information he wants. The chances are that he would be wasting his time to specify the exact format and classification structure of the information he believes he needs; it would simply not be practicable to provide him with it.

The computer has changed all that. One dare not any more ask for "everything that can readily be provided." A fantastic variety of cross classifications and different formats can usually be provided, and also usually at reasonable cost. That is, until someone has to be paid for looking at the voluminous reports. Again, as in the administrative routines, the past habits of thought have to be discarded and a new and more precise specification of the needs for information has to be developed. The exact nature and form of the information might dictate some of the characteristics of the system: how files are kept and updated, which pieces of information can be linked together. For example, it might be possible to set up some separate files for special purposes and thereby to simplify both the content of the special file and that of the general files. But this usually cannot be done economically if linkages between the information in the special file and that in the general file are important. They might be important to management intelligence but not to the administrative routine. Hence, the informational demands of top management must also be recognized in constructing the information processing system.

A simple illustrative principle might be cited. It is not infrequent in both government and industry that an administrator works laboriously over daily, weekly, or monthly reports on the operation of his shop. What is he looking for? Whatever it is, the chances are that a computer can find it faster, better, and less expensively than he can, provided that he can define what it is. He might list a whole hierarchy of criteria. He wants to know of a set of workload statistics when any of the following occur:

1. A decline in any one day of 5% in an operation he rates as large or important -- and he specifies which operations these are;

2. A daily decline of 10% in a small or less important operation;

3. A decline of 3% in weekly totals for the larger operations;

4. A decline of 5% in weekly totals for the smaller ones;

5. An increase in a single day of 8% in a large operation;

6. An increase in a single day of 12% in a small operation; and so on, taking account of previous years' experience, items that bear special watching at this time, etc.

What if none of these situations develop? Then it should suffice to assure the executive that the appropriate search has been made and that there is nothing in the last set of reports that warrants his attention.

What if the computer prints out some cases of data to be called to the attention of the manager? Then it becomes necessary to learn what he would do with them. If he wants associated cost data, or ratios to previous weeks or months, or percentages of some assigned standard work unit, the computer should be required to derive these facts in precisely the desired format. With the ability of computers to take percentages, it hardly pays the manager to copy numbers from a computer output tape so he can divide with a soft pencil and paper. Moreover, if he would like to call some of these data to the attention of others in his organization the computer can write the required transmittals, and even make a few nasty comments, if this is the way the manager wants to do business.

This leads into a part of management science that will only be mentioned here, since it could appropriately be made the subject of a separate seminar. Much interesting and elegant technique has recently been developed for the alleged purpose of helping the manager make his decisions: linear programming, quadratic programming, queueing theory, game theory, inventory control models, and lots of variants and extensions of these techniques. They seem to apply usefully only to limited

ranges of actual managerial problems. However, if any of these techniques are to be used, the computer can usually reduce the computational burdens of their utilization. Certainly, if the necessary data are already in the high-speed part of the machines for another reason, there is little merit in dumping them out so that they can be reintroduced as a separate problem, or solved by hand.

Throughout this paper, it has been necessary repeatedly to lay a difficult problem at the feet of the executive officer of an organization and ask him to solve it. In general, we have offered little advice about the methods and techniques he ought to use in the search for these solutions. This is not to be traced to the natural shyness of computer technicians. It is because they do not know how to solve them either.

Among other difficulties is the presence of a meta-problem of the management sciences. It is the general absence of a suitable measure of effectiveness of the work of an organization. A profit making organization at least starts with some kind of presumption of a desire to make as large a profit as possible. The situation is never this simple; there are always a number of restrictions within which the activity should take place. Some of them are legal, some are ethical, others are merely personnel preferences of those in control of the operation. The decision not to hire women in an enterprise might be based upon shrewd appraisal of the disadvantages of having them in the particular kind of organization and operation, or it might merely reflect the belief that the place of woman is in the home.

Most government agencies do not even have a formal measure of profit to guide them. There is no accepted accounting which purports to measure their success rates. The expenditure of funds can always be cited, but this is a measure of the resources brought to bear. What was accomplished with it? Without some measure of the effectiveness of the operation, without a suitable accounting system, how can the executive decide whether to formalize his operating procedures for an added degree of automation or not? How can he decide how much specialized information he should have provided for his own use? He is not going to make a profit in any case.

There is some belief among theorists that these matters would become more manageable if there were some theoretical underpinnings to the business of determining the "best behavior" in some useful sense. The technical journals continue to spread articles, some of them quite technical, on optimizing models of all sorts. The general approach is to invite the manager to state the limits within which the game must be played and then to seek a high rating either on a single scale or on a combination of two or more scales which purport to measure the effectiveness of an operation. There is no guarantee of salvation here for the executive who still has all of his old problems and now also has a large and expensive computer being deposited on his already busy desk.

The best that can be offered is a promise that the advanced techniques being proposed will be less onerous to use if the organization has a computer, and much less so if the required information is already within the mechanized part of the operation.

If there has been any progress along this line it is in the insistence that all costs of the system be reckoned with: the costs of generating the information in the first place, all of the processing costs, the cost of the executive time in making the decisions. When realistic costs are assigned to these processes, the goal is often restated. It is not necessary to find the "very best" answer, because of the large demands for accurate and up-to-date information. Rather the concept of a "good enough" answer takes an important place. We return, then, to the idea of a convenient solution, in much the same form as it appears in the search for a set of "good enough" rules that will allow a computer to take over a routine operation, even though a "better" outcome might be obtained at higher cost.

A specific example of a system in which formal rules are introduced (suitable for programming onto a computer) will be given. The importance of considering all of the costs will be evident. Indeed, it will be apparent that undesirable consequences might well follow from the omission of costs of checking identifications or costs of delaying an action. (See Figure 2, below.)

Figure 2.--The Matching Problem*

Decide a MATCH	Decide a MATCH	Hold up All Actions	Decide a NONMATCH	Decide a NONMATCH
Proceed without checks	Also Institute checks	and Institute checks	Also Institute checks	Proceed without checks

The objective: Not to minimize errors, but to minimize costs.

* Based on an unpublished paper of B. J. Tepping, National Analysts, Inc. Used by permission of Dr. Tepping.

((Note on Figure 2: The "Matching Problem" is one frequently met. Its general form is as follows: A file exists which contains information about a number of cases. A transaction of some kind occurs, creating a need to know whether the individual now under consideration is included among those already in the file, and, if so, which of the cases in the file corresponds to the case represented in the present transaction.

What should be the objective of the matching routine? It might seem intuitively that the best rule is to require that the identifying information on the transaction (say, the person's name and address) match those in the file exactly or we shall rule a non-match. This is a very strict rule: it should produce an extremely low rate of erroneous matches (cases where the decision to associate the new transaction with a case in the existing file is in error. This could arise, for example, where there are two Robert Joneses living at the same address). But there would be a high rate of erroneous nonmatches: cases in which it was ruled that the individual in the new transaction was not contained in the existing file, but in which this conclusion was in error. For example, Mary Jane Smith of 123 Maple Street might also write her name Mary Smith or Mary J. Smith. She might also turn up as Mrs. George A. Smith, or even as Mrs. G. Alfred Smith after her husband has had a raise. If any of these variants is used, the exact match requirement would cause another file to be set up as though all of these were different people. A set of "matching rules" can be devised which will require less than perfect coincidence of each character. Even fairly loose rules will have difficulties in making matches if some of the variations above are accompanied by rewriting the address as well as the name: e.g. corner of Maple and Pine, or Hightower Apts.

It is evident that an enormous number of sets of matching rules could be written. What should be the criterion of selection of the "best set" in a given case? The immediate temptation is to choose the set that produces the smallest number of errors, adding the erroneous matches and the erroneous non-matches. This, it will be argued, is the correct criterion only in the infrequently met case where the cost of an erroneous match is the same as the cost of an erroneous non-match! More generally, a desirable objective would seem to be so to construct a set of matching rules that the costs of all errors (of both kinds taken together) should be minimized.

Dr. Benjamin J. Tepping of National Analysts, Inc., in an unpublished paper, describes a formal model which presents a set of procedures (say five, in this case) for dealing with situations with respect to matching:

1. The specific rules under consideration define a class of coincidences which will lead to the conclusion that the name and address on the new transaction is sufficiently close to one in the file that we shall make the match and then proceed with the business of the organization in the same manner as though the coincidence were perfect.

2. The rules also define a looser class of coincidences, which still strongly suggest that there is a match, but in which there is sufficient doubt about it that a check is instituted to determine whether there really is a match; but the business of the organization proceeds as though there is a match. The "checks" might consist of matching the handwriting on the manuscript documents, writing an inquiry to determine whether the Mary Jane Smith is really Mary J. Smith and whether 123 Maple Street is on the corner of Pine, etc. Such procedures increase the costs of the operation, but enough additional information might be obtained to meet the requirements of the first class enumerated above.

There are cases symmetric to these:

5. Decide a nonmatch and proceed with the business.

4. Decide a nonmatch, but institute checks to be sure of this choice.

3. Finally, a set of rules can be written in which there is sufficient doubt about the matching status of a new transaction that *no* action at all is taken except to seek more information. On the basis of the additional data, it is hoped, the case can be moved at least into classes 2 or 4, which allow the business to proceed, in parallel with further inquiries about the case. It would be much better, of course, if the additional information would lead to the assignment, by the matching rules, into classes 1 or 5 which would amount to a "final" decision about the case in the absence of complaints or other new evidence bearing on the identity of the persons in the new transaction.

Such a set of matching rules will lead to errors in cases 1, 2, 4, and 5, and *costs*, for the checking of identities in 2, 3, and 4, and *costs* of erroneous matches in 1 and 2, and of the erroneous nonmatches in 4 and 5.

The Tepping model constructs formal apparatus for translating the *probabilities* of errors in each class and the *costs* of errors in each (plus the cost of waiting in class 3) into an analysis of the total costs to be minimized.

Thus, as frequently met, the task of matching a piece of incoming business against an existing file (which might or might not contain information about the individual) proves to be subtle and difficult.

Modern computers have introduced a great freedom of choice in the construction of matching rules. The theoretical apparatus suggested by Dr. Tepping shows that a rational approach is possible, even when the costs have to be estimated. Specifically, the model tells us when to stop looking for further confirmation of the matching decision. Where the costs of errors are high or where there is large volume of such matching inquiries, a detailed research effort may well be justified by the eventual savings of time and other resources.))

THE FEASIBILITY STUDY FOR BUSINESS APPLICATION

Charles F. Kiefer

Charles F. Kiefer is Executive Director of the Management Operations Staff, Agricultural Economics, of the Department of Agriculture. Previously, he was Assistant to the Deputy Administrator of the former Commodity Stabilization Service, in which capacity he had a large part in the conduct of feasibility and systems studies, and in the installation and organization of data processing activities.

It is appropriate at this point in this seminar to begin to think through together a few of the issues which must be overcome in a practical way before one can even approach the problem of actually acquiring and managing an ADP installation. Of course, it is entirely possible that top management knows enough about the processes and details of its business that it can decide -- <u>he</u> can decide -- to order and acquire a computer or a sizeable punch-card installation to meet his present and prospective needs -- without much preliminary analysis, discussion, and consideration.

There have been a number of firms which have taken this route to operations improvement. Basically, these managements must have said to themselves: we know our operations quite well; we know what the equipment is generally capable of doing; we are therefore deciding to get one of these machines as soon as possible. Then we can learn more about it, learn how to use it effectively, and outstrip our competition by being able faster to have more timely information at our disposal and make wiser decisions.

It is undeniable under the assumptions implied in the foregoing decision that a formal feasibility study is unnecessary and uneconomic. There is, moreover, a certain high degree of faith in the making of such a decision in contrast with having in hand the organized facts. In a public civil agency, it is doubtful indeed whether the preceding outline of the route to a computer would be successful. Not only is the literature on the subject of the wisdom of conducting a feasibility study impressively persuasive (so are the <u>cases</u> where it has been bypassed!), but the guide lines in this area are quite clear. In this connection, I refer you to releases from the Office of the Comptroller General of the United States, the reports of the Inter-Agency ADP Committee, directives within your own agency, and the hearings before committees of the Congress. You as top administrators have long since come to grips with the exigencies of the budgetary process, and you recognize the steps in the justification of a proposed course of action.

And so -- it is central to this discussion that the decision by a public civil agency to acquire an ADP configuration to perform business management-type operations or research-type operations follows rather than precedes a careful study of all of the revelant factors in the problem situation. This careful study is called the feasibility study.

It is desirable to recognize here the growing opinion that the word "feasibility" be replaced by the word "systems." I do not propose to analyze the many reasons for this opinion. It is, however, more than the mere use of words.

Feasibility means -- capable of being done, executed, or effected -- reasonable, practicable. Feasibility of what? Feasibility of a whole system for performing a program of work through a computer? or a segment thereof? or an individual part of or an application within the system?

Let me illustrate. First, I will define a system simply as a complex of interdependent ideas, principles, methods, and procedures said to be unified, connected, managed, and performed by people and equipment to achieve stated objectives. Now for the illustration of a simple inventory system. Let us say it is a new soap product called "scrum." We all recognize the probability that the product is produced through the use of equipment. After it is produced, the "scrum" is acquired into inventory. It must be maintained. It must be moved -- by truck, air, or rail. It must be sold. It must be invoiced both for storage and for sales. It must be re-ordered. Materials must be purchased, rents paid, salaries paid. Well -- I could go on. But you are already ahead of me. The production process is mechanized. The distribution and management processes may not be. Can they be mechanized? Is it practicable? Why even think about it?

Well, sales of "scrum" have increased to the point where a large clerical force is needed, possibly in several cities; there is a large volume of repetitive clerical operations; the use of different formulas of "scrum" for different cleaning purposes has produced a number of complex computational problems. With the increasing demand by management for faster sales and inventory information and the need to schedule both production and distribution operations more effectively, a harassed middle as well as top management seeks relief!

Possibly the sales invoicing operation is the segment to study -- with an eye towards a computer? Or the re-ordering of supplies? Or the payroll? Which one of the many segments? Why not all? In this overly simplified illustration lies the problem and the need for someone to define the problem.

It is my view that it is highly desirable and vitally necessary to have a total systems concept and a total systems approach early in the planning for the use of automatic and electronic data processing equipment.

It has been, and undoubtedly will continue to be, attractively enticing <u>separately</u> to commit payroll operations to the computer or to mechanize the preparation of monthly or quarterly invoices to customers or to speed up the preparation of monthly and other periodic reports. To do operations or applications of this type will provide, it has been said, an early pay-out, needed experience, and a dramatic demonstration of the capability and versatility of the computer. You can readily see how these ideas and experiences could occur under my illustration of "scrum" operations. I suspect that a considerable portion of the disillusionment with the computer that has occurred with some users has been in those instances where a total systems approach has been somewhat lacking.

A total systems approach is concerned with the program or programs of work of the agency. The definitions found in the annual budgetary submissions to the Congress will be a useful starting point. The total systems approach will be concerned with the methods and processes by which these programs are carried out, <u>and the operating costs thereof</u>, both administrative and program! This approach will be concerned also with the views of those, both inside and outside the agency, who assert that there are opportunities at hand to improve agency systems and methods or who quietly insist there is a need for a change in the performance of the program as a whole, and give their reasons. The systems approach will be vitally concerned with the short and longer run needs of top management, the operations of the accounting and auditing divisions and branches, and the day-to-day work of the operating or subject-matter bureaus and divisions.

The purpose of the feasibility or systems study is to discover whether top management can justify the acquisition of an ADP system. The study will outline how the analysis was performed, the scope of the considerations, the cost-benefit aspects expected to be realized, the timing of the analysis and of future involvement, the specifications of both the problem and of the required equipment (whether conventional punch-card or EDP). It will include a detailed description of the agency's present system, including the cost of operation, and it will address itself to all practical aspects of the problems sought to be solved by the use of the equipment recommended.

It will be concerned with work processes which involve complex computations or which require a large clerical force performing a large volume of repetitive routine-like operations. Notice will be given also to timing of reports needed by top management and the scheduling required to meet these needs. The Budget Bureau has issued a useful publication from one of the Task Forces of the Inter-Agency Committee on ADP. I urge you to secure a copy and read it.

Let it be understood that the making of a feasibility study itself is comparatively expensive, depending on the scope of the problem(s) identified and analyzed. I have with me several feasibility studies. The cost of making these studies ranges from $8,000 to over $40,000.

And it is important to define your terms of cost even in the making of the feasibility study. It can and does happen that the original feasibility study requires amending and up-dating as times goes by.

Before arranging for the conduct of feasibility studies, some <u>reconnaissance</u> in the area of systems review and data processing equipment is undertaken. This may be a reason why many of you are participating in this seminar. This reconnaissance, authorized by top management, involves the making of a "judgment analysis," either verbal or written, of operating program data of the agency <u>to determine whether ADP and the computer should be investigated</u>. I do not have copies of this type of analysis with me, although the files of correspondence in our agency add up to the obvious specifications of such an analysis. Ideally, I suppose, this "judgment analysis" would have the following characteristics:

1. It could be an individual or a staff study, depending on the circumstances;

2. It would involve preliminary identification and inventory of the probable areas of agency work where research-type or business-type operations improvement is needed and where the potentiality of the computer <u>seems to have</u> application;

3. It would involve a hard-headed analysis of the present organization and its people to determine not only its capability to make the required subsequent studies, but also to manage and to operate existing programs under a conceivably new technology with little impairment to short-run objectives, and significant achievement of longer-run objectives. The exigencies of the budgetary process must be observed.

4. This initial "judgment analysis" may lay the background for further investigation by the agency; <u>or</u> it may recommend a program of systems and feasibility studies within the agency, a later result of which may be the acquisition of a computer; another result might be that no computer is needed, but that changes can be made to achieve agency objectives sooner, faster, better — short of a computer.

5. If the "judgment analysis" recommends an active program of work in the systems area, or in the performance of agency internal and external reporting, or in research processes, or in specific operating and statistical service areas, or in some other appropriate combination approximating a systems approach, it must:

(a) Outline the objectives of the program of work. This would identify the areas for study, the operating improvement sought in service, and other objectives of a program character associated with the purposes of the law authorizing the program(s) run by the agency.

(b) Delegate the responsibilities for leadership, coordination, and further study.

(c) Provide for control and reporting of significant endeavor.

(d) Indicate some of the hoped-for results, for the middle <u>and for the long run</u>.

(e) Recognize the cost aspects and other alternative consequences of further actions.

This course of action is, in effect, a feasibility study of the advisability of making further feasibility studies. Some of the results would be: (1) To keep on doing what we are doing in the way we are doing it. No further investigation seems warranted; (2) Request or secure funds to undertake more comprehensive analyses. Wait and see a little longer; (3) From existing <u>scarce</u> resources, allocate the necessary funds to embark upon an ADP-EDP program, and provide for the <u>estimated</u> requirements <u>envisioned</u> in the "judgment analysis" in the annual budget estimates for the succeeding fiscal year.

Let us now assume that the agency head or his deputy wants to make <u>and</u> is able to make the decision to embark upon a program of work in the systems and feasibility study field where a computer may eventually be required; he is personally convinced that the effort will be worth the cost; he is reasonably knowledgeable on the systems approach to the orderly utilization of a computer. In the days immediately prior to the moment of decision, other decisions need to be made as to:

1. Who is responsible for coordination, subsequent decision-making within broadly established policies, and for follow-through;

2. The utilization of present organization and staff to do the analytical job -- this is a very important decision;

3. The creation of a committee, task force, or special staff broadly representative of the agency's principal functions, comprised of individuals who know the agency's policy, work operations, history, and objectives;

4. The need for informal or formal Budget Bureau and Congressional Committee understandings;

5. The training needs of the agency;

6. The need to enlist the understanding and support of key officials throughout the agency -- this is of the highest importance;

7. The adequacy of organized documentation within the agency as to what and how the agency's work is being done; what about forms and records management?

8. The orientation of employees in general in the agency;

9. The approach or approaches to be followed:

   (a) Ordering a computer,

   (b) Systems and feasibility studies,

   (c) Supporting project or applications studies;

10. Ultimate approaches to equipment acquisition and present and future relations with equipment manufacturers. (I have with me a handout which depicts an approach our agency has used effectively in this competitive area);

11. Time-table of the work plan for the foreseeable future.

Some of these points may have already been covered by recommendation in the reconnaissance study. Even so, decisions are needed in the final "wrap-up" by the agency head. It is a good thing to have these decisions well documented!

Implied in all the foregoing discussion has been the assumption of well-qualified internal agency leadership and supporting personnel to do this work in systems review, operations analysis and research, and feasibility studies. These employees would have admixtures of training and experience in management engineering, in computer and punch card operations, in systems and procedures, in accounting and auditing, in communications, and in writing clear, concise English. Obviously, some systematizing of effort is required. It will not be long before the heads of the operating, or functional or staff or subject-matter divisions begin to display the relative importance they have assigned to this work -- whether it is considered a "fad" and a "binge" or a real opportunity to accomplish and achieve agency goals! Another way of quickly assessing this developing climate is by discussion with the sales representatives of the machine suppliers. It is most frequently what they do <u>not</u> say that is significant.

If you decide to use a management consulting firm in any capacity, it is wise to consult with your Department Solicitor or General Counsel. There are undoubtedly real advantages to be gained from an "outside" advisory viewpoint, depending upon the circumstances prevailing in the particular agency. It is well to keep in mind that such a firm, while otherwise eminently qualified to perform the services it advertises, is a stranger to your people and probably to your program. It is a good idea to have concretely in mind and on paper precisely and specifically what is desired from the "outside firm." The costs of these services range from $200 to $400 per day per man to upwards of $25,000 for a contract, again depending on the requirements and duration.

Let us now proceed to the conduct of the feasibility or systems study itself. I shall seek to illustrate the points in this discussion by reference to the work of the Commodity Stabilization Service.

Let us assume that your top management, after consultation with experienced personnel in the Budget Bureau, in the General Accounting Office, in the Bureau of Standards, in the Department of Defense, and elsewhere in and out of government, has made the decision to embark upon a program of work in the systems and feasibility study field where a computer may eventually be required. Let us assume that an agency committee or task force or both have been created, and that collateral action is being taken along the line covering the points referred to the "judgment analysis." Finally, let us assume that the committee or task force or both have formulated and reduced to writing the systems concepts applicable to the agency's program of work within which authorized feasibility or systems studies will be carried out. This has both taken time to do and has been approved by the agency head or his designee. Keep in mind that all of this work costs money. It may well be that the committee or task force or both will conduct the feasibility studies. This makes sense in a small agency. In a larger agency, however, it is desirable to decentralize the conduct of authorized feasibility studies so as to convoke as many of the best brains as possible within the agency in the consideration and selection of areas of work warranting further detailed investigation. Obviously, the people closest to the site where the work is performed know more about the work than the people at headquarters. Yet it is wise to keep the headquarters staff closely in touch with field studies and to assist wherever possible. The questions of centralization vs. decentralization, of equipment compatibility, are difficult but not insurmountable.

I have already noted the purpose and principal ingredients of the study. But it is of the highest importance, after selecting the area or areas for study, that consideration be given (1) to the present and future needs of top management in decision-making, and (2) the manner in which these needs are presently met. In other words, where does management get its information, how is it created, what form is it in, what is its timing, what are the administrative, legal, and audit requirements surrounding its transmission, maintenance, and destruction.

These are the ingredients of input analysis. There are other considerations on the output side dealing with the uses and users of the information. It is a good idea to delve deeply within the output area, using flow charts, forms and records logs, and available cost data for separate phases of the area of work under study.

At this point, keeping the system concept in mind and having the details of the methods and processes being used in present operations as well as present costs, as defined in the authorized study outline, time is taken for a careful analysis of the data collected. Here is an early source of pay-out in the streamlining of work procedures and the elimination or simplification of forms and reports. It is conceivable that the work done up to this point should pay for itself.

But we are investigating whether an ADP system can be justified. Therefore, having in mind the general capabilities of a computer, the question becomes -- how shall the agency's work (as defined in the study authorization) be carried out through a computer? This means visualizing how the basic data will originate and flow, what happens to forms and reports, and to exceptions to the visualized routine. This means analyzing the volume of data to be processed, the processing time cycles, the persons within and outside the agency who may be affected by the possible changes, the best estimate of projected costs, and the best judgment of tangible and intangible benefits to be secured from an ADP system. It can well be understood that an analysis of each of the foregoing areas may lead to the conclusion that an ADP system cannot be justified. Of course, the opposite conclusion is frequently reached. I have with me illustrations of studies of both types of situations in my agency which we can discuss further if you wish.

Although there is no fixed rule regarding the next step in the study, there is a certain logic apparent. We have already identified and defined our problem or problems. We have analyzed all facets of the problem situation, including costs and collateral benefits. We have itemized our reasonable needs and the alternate methods available to meet these needs. We have reviewed the basic purposes and goals of the agency, and we have, let us say, determined that a computer will do a better job than is currently being done for the agency. We have already cleaned up portions of the present system on the way to the practical written development of specifications to perform the same work under a new system. We have consulted with key personnel inside the agency as well as inside the Department. We have also consulted with our clientele. The outlook is promising, indeed.

It is logical, then, to reduce the foregoing analysis to summary form, supported by the details, for the purpose of securing from the manufacturers of ADP equipment their ideas as to what equipment system will do the job best. This may involve a briefing session prior to the release of the invitation to bid to do the job. A reasonable time is provided for written replies. This is also a highly educational part of the study.

This is where the machine supplier selects from his equipment array on the market or coming on the market that combination of machinery and process that, in his judgment, will do the best job and accomplish agency purposes.

The review of these responses from manufacturers is a compelling task. It requires considerable knowledge of computer capabilities and allied equipment. One must be able again to visualize how the agency's work as defined in the study will be performed under each response. On the agency side, there are considerations of data volumes, processing time requirements, and costs. On the equipment side, there are considerations of equipment flexibility, compatibility, reliability, speed, error incidence and control, and cost, whether rented or purchased.

At this point, further extended analysis and consideration is required not only to select the equipment best suited to the job but to develop recommendations as to personnel needs and training, site preparation, further system design, programming, testing, parallel operations, and the concurrent development of instructions, forms, and reports.

When this analysis is complete, it should be possible to conclude the feasibility study with recommendations to top management covering the need for change in present methods of operations, the manner by which the needed changes can be effected, the computer system available and selected, the full development of present and future costs, savings, and benefits, and the foreseeable consequences, favorable and unfavorable, arising from adoption of the recommendations. There should also be a time-table for future planning and action.

I am almost finished. What I have described can take as long as a year to complete. What lies ahead after a decision favorable to computer acquisition is often in excess of 18 months. But if the feasibility study is carefully prepared, it is a great economy for top management and, usually, can only be rejected for over-riding program or policy reasons. The study must show positively (1) that the gains and benefits are worth the cost, (2) where the break-even point in total investment is, (3) that no other way to do the job is as cheap, and (4) that the relevant experience of the agency and of other agencies supports the venture, that the enterprise of adopting a new system is feasible.

Let me add a word about personnel affected, by reading a comment on a recent study by the research workers at the University of Michigan.

"Government and industry are stepping up joint studies of the impact of office automation, the so-called 'wave of the future,' on company employees.

"A new study, part of the automation research project of the University of Michigan Labor and Industrial Relations Center, Ann Arbor, Michigan, puts the spotlight on the Midwestern home office of a medium-sized insurance firm. It suggests that the process of installing a computer, and adapting work methods to it, may cause noticeable though presumably temporary employe dissatisfaction unless management handles the conversion skillfully.

"The study established that installation of a computer -- in this case, a medium-scaled IBM 650 -- is not a radical or extensive enough operation to cause a substantial reversal or acceleration of existing trends in work environment and job satisfaction.

"No large technological changes developed in the study's pilot company.

"The study indicated that installation of the computer affected the work environment of a number of employes, that most of its effects were those that employes desired, and that the computer installation was liked more often than disliked.

"Employes in the computer area, which gained work tasks as a result of the installation, and employes of the unaffected departments liked the net changes more than did employes of the other affected departments, which had lost work tasks and had been required to adjust to some new work methods, according to the study."

A final word -- the original feasibility study will be widely read, may be wrong in places, will undoubtedly require revisions in the light of experience -- but it will stand as a bench mark in the progress of your agency if it is most nearly right in its conclusions. It is just the beginning of a period of exciting adventure in public administration.

PART 5

ADP SYSTEM IMPLEMENTATION AND OPERATION

PLANNING AND SCHEDULING BEFORE ACTUAL INSTALLATION OF AN ELECTRONIC DATA PROCESSING SYSTEM

George F. Stickney

George F. Stickney is Head of the Fiscal Service Operations and Methods Staff, Office of the Fiscal Assistant Secretary of the Treasury Department. He has been responsible for the conduct of many significant programs for procedural improvement which have resulted in substantial economies as well as increased efficiency in operations, such as a mechanized program for handling deposits for withheld taxes and an integrated electronic system for the payment and reconciliation of Government checks.

I appreciate the opportunity to be with you today to discuss the important subject of planning and scheduling for EDP equipment before actual installation. Discussions like this seminar on Data Processing for Federal Executives of the Department of Agriculture Graduate School provide a means of exploring the subject from a number of viewpoints, and render a real contribution in this field. My part in your program will relate primarily to what we have learned about preparing for the installation of an electronic data processing system.

Review of Initial Study

I ask your indulgence if at times it appears that I am digressing from the subject because I believe it is necessary during this phase of an installation to take a look backwards to review the initial study and what has been accomplished, as well as to look into the future. To start planning for actual installation without reviewing the past in relation to the future might well make the difference between success and failure.

Experimentation

As you know, the history of utilizing electronic data processing equipment, short though it may be, is filled with thousands of man years of experience. This experience covers the entire area of data processing involving the development of systems to harness electronic power to carry out many complex problems. The results, through experimentation--and it truly has been experimentation for most everyone concerned--have been most gratifying when viewed as a whole. Some of the more important things we have learned are:

1. Processing cost is less,

2. Quality of output data is greatly improved,

3. There are greater potentials for service to management, and

4. To be successful in achieving these results it is necessary to "rethink" the existing system from beginning to end.

Let us first consider the last two points concerning service to management and the need for rethinking the whole job.

Improvement in Service to Management

The development of systems and procedures for service to management has received increasing recognition throughout the Federal Government during the past 10 to 12 years. The problems which challenge management of the Federal Government today are unlimited. No longer are these activities limited to services such as delivery of mail, supplying coin and currency, law enforcement activities, collection of customs and taxes, assistance in improving the agricultural efforts of the country, and conservation of our natural resources. We still have these and, I might add, their related problems have kept pace with the ever expanding growth of our country. But, in addition, our Government is currently engaged in practically every type of activity known to private industry and many which are unique to Government alone.

Management responsibilities in the network of complex operations of the Federal Government are likewise increasing. They range from installation supervisors to those who exercise leadership at the highest levels.

Experience has taught us that properly planned EDP installations have resulted in greatly improved services to all levels of management. These advantages result in better and more useful (not necessarily more in quantity) financial management data. Later we will discuss the potentials for the development of automatic decision-making processes during the period of systems changes.

Need for Rethinking

This leads me to the next point in our review of what has been learned from experience. It is, I believe, an almost tragic aspect of electronic data processing systems development that application after application can be found where the electronic system parallels the former system. The necessity for "rethinking" the entire process step by step has not to date received sufficient emphasis. Possibly this lack of emphasis can be traced to the absence of the so-called profit motive in Government operations and some resistance to change. This is superficial thinking, since we in Government should exercise the highest sense of responsibility for the control and efficient use of all the

public resources placed at our disposal. I am convinced that, given the proper emphasis, the matter of "rethinking" the complete system will pay handsome dividends in terms of reduced operating costs and better service to management and, in turn of course, to the public.

The question of how best to focus emphasis on this matter of "rethinking" is a subject which deserves a great deal more time than is allotted for this seminar. However, there is one important ingredient that must be found in any organization before the "rethinking" process will become an actuality and that is the accountant, auditor, management analyst, and all classes of systems and methods examiners must understand the needs and problems of management at each level. They must gain this knowledge by participating as members of the management team--as active advisers to management at each level of responsibility. In turn, Government managers and top-level executives and their staffs must instill in the management and systems leaders within their organization a zeal for accomplishing what sometimes appears as impossible and, I might add, " impractical." This does not mean that executives should encourage non-compliance with requirements which are characteristic of a Government organization managed in part through the discipline of appropriations and statutory requirements governing the purposes for which public funds may be expended. All too often the path of least resistance becomes the rule. This is borne out by the fact that so far as I know all rivers run down hill. That is, all but one--the Chicago River--it runs up hill. It would be interesting to learn what went into the process of thinking out the system for this project.

Before closing my remarks on the importance of "rethinking" I would remind each of you that the present check payment and reconciliation program in the Office of the Treasurer of the United States happens to be an example of what I have been attempting to bring out. Here is a case where it could have been relatively easy to convert each step of procedure in the Federal Reserve Banks, the Treasury Department, and the General Accounting Office to electronic processes. While the non-electronic methods used prior to 1956 made the separation of these various functions not only possible but practical, the adoption of EDP opened up the possibility of integrating all functions under one consolidated process. This integration never could have been accomplished if top management had not encouraged the systems team to "rethink" the entire process of clearing, paying, and reconciling Government checks. There are some other good examples throughout the Government, but not as many as there should be.

Systems Design

The planning and preparing for actual installation of EDP has many facets. The program will differ radically depending on how much and what type, if any, of systems design was included in the feasibility study. You have held discussions on how a feasibility study should be undertaken

and carried out. I would like to take a few minutes to review this subject with you inasmuch as it is very important in planning for installation.

The most important point to consider can be illustrated by asking the question "Just where do you stop in the matter of systems development in carrying out a feasibility study?" Well, like most things in life, this becomes a matter of judgment. However, because decisions are influenced--yes, are made--from the results of feasibility studies, it follows that such studies must be based on facts. In my opinion, these facts have to be based on considerable knowledge and understanding of the proposed system. I have seen feasibility studies recommending that EDP equipment, involving the outlay of substantial sums of money, be acquired with very little reference to the system to be involved. Probably because my experience in the Treasury Department has been such that the cost of operations always has been the prime consideration, I feel that as a very minimum the logic of the proposed system should be completely defined and considerable research done to determine what the new system will cost before a decision can be made to acquire EDP equipment. I would readily agree that, if cost is subordinate, the feasibility study does not necessarily have to encompass the development of the system logic in great detail.

Establishment of a Plan of Operation--EDP Committee

Once top management has decided to acquire EDP equipment as a result of a careful feasibility study, it will find itself in about the same position as a commercial concern which has decided to build a new manufacturing plant. What remains to be done and, more important, to be successfully carried out, are plans for selecting the EDP organization, developing a time table and target dates, selecting and training the programming group, designing of the system (if this has not been done by the group making the feasibility study), planning initial parallel operations, and initiating actual operations. This places a responsibility on top management for careful preparation, selection of qualified people, and effective control over the progress being made by the EDP group. It requires attention and the ability to recognize when and if plans should be modified. Last but not least, it requires cooperation at all levels of management and on the part of those whose work will be affected. It creates problems, the solution to which may mean the difference between success and failure. In summary, experience teaches us that it is necessary for top management to exercise control over the installation program and it is essential that it be kept informed about the progress being made. Based on experience gained to date, I feel that one of the best ways to accomplish this is to form a committee including, among others, one or two of the people engaged in the original feasibility study. Such a group provides a communications link between top management and the EDP group. Moreover, such an

arrangement probably can best serve management by being in the position of focusing attention on the objectives which motivated the undertaking of the feasibility study. By this I refer to whether cost or quality of service was the important criteria. Also, such a group can make sure that decisions of policy are not disregarded or are being delayed if they are required. Another important point, the nature of the work of an EDP Committee is such that it will require practically the full time of its members from its inception until the installation is operational. (Figure 1 on page 148 illustrates a typical EDP organization.)

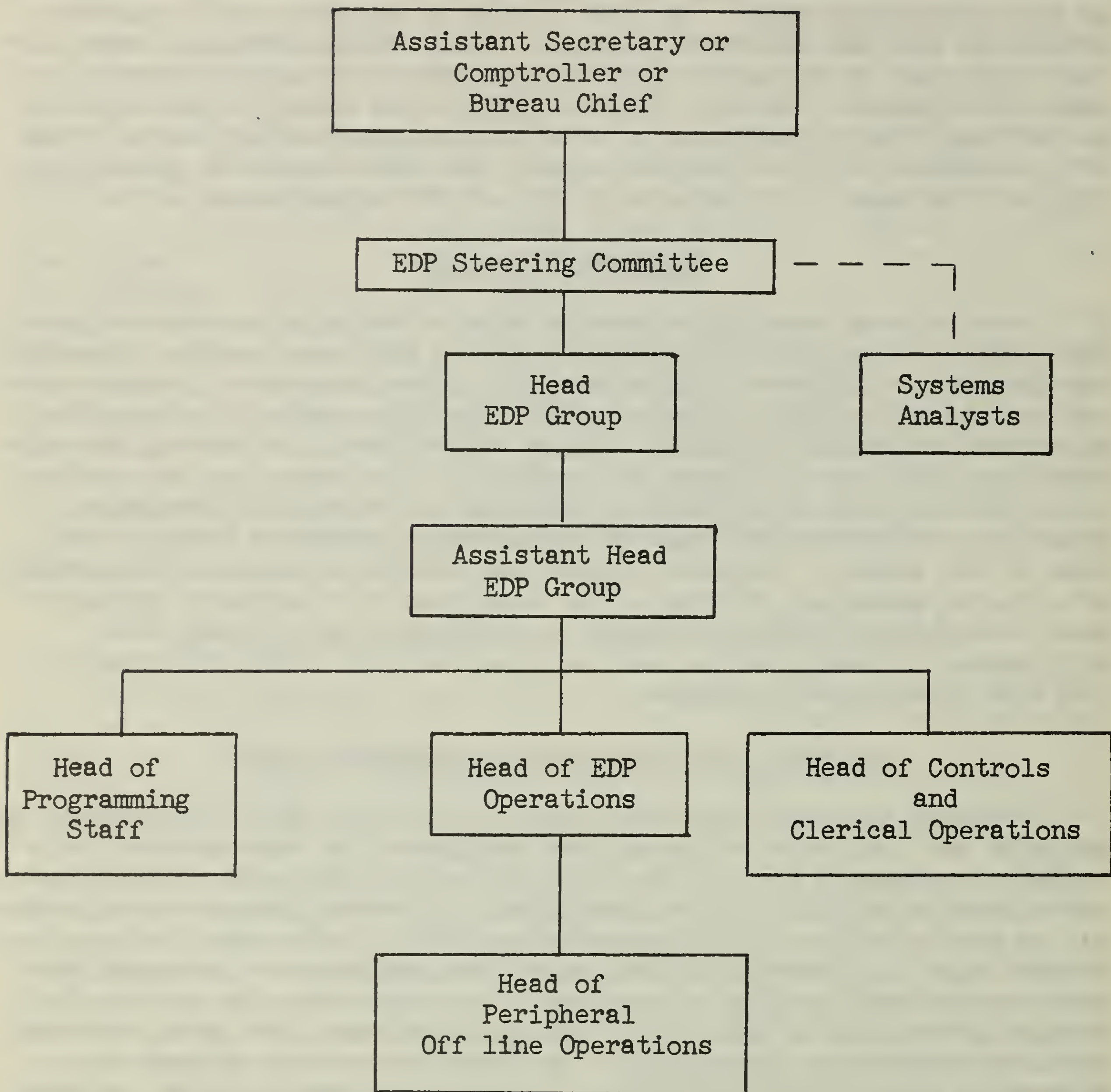
Displaced Personnel

Most matters requiring attention after a decision is made to install EDP equipment must, of necessity, be carried out concurrently. However, the problem of displaced people should be given priority. In many cases, the acquisition of EDP equipment will ultimately result in substantial reduction of present personnel. Every effort should be made at the earliest possible date to inform all employees to be affected, as well as those who may have an interest in the plan to install EDP equipment. Experience indicates that such an announcement should be made by the head of the agency. Progress reports and an active program of placement of affected personnel will make the transition to electronic equipment more successful. Personnel should be encouraged to indicate their willingness to train for the new job opportunities which will open up for them under the new program.

Selection and Training of Programming Staff

Probably the most important, surely one of the first steps after the decision has been made to install EDP equipment, is the selection of a capable programming group. The capabilities of this group will influence the degree of success of the undertaking. I cannot place too much emphasis on the selection of the programming group. I also have a strong conviction that the operations group should be comprised of personnel who have proven their ability to program. I recognize that there is a difference of opinion in this area. You may have heard the opposite theory advanced in previous discussions. Probably the test of whether the operations group should be selected from the programming group can best be made from a look at the applications to be undertaken by the installation. If it is determined that the installation is to be involved in numerous different machine runs of short duration and deadlines for completing such runs are not important, then the operating group could in all probability be selected without too much emphasis on their knowledge of programming. However, when deadlines are important and the installation is engaged in varied long machine runs, experience shows that knowledge of programming is an extremely important resource to the operations group.

Figure 1.--



Assistant Secretary or Comptroller or Bureau Chief
EDP Steering Committee
Head EDP Group
Systems Analysts
Assistant Head EDP Group
Head of Programming Staff
Head of EDP Operations
Head of Controls and Clerical Operations
Head of Peripheral Off line Operations

The selection of the programming group should be based primarily on quality. You may wonder why I feel that quality is paramount. Well, it is because the work involved is exacting and requires intellectual ability. The degree of success of the application will rest on the efficiency of the program. The machine is entirely dependent on instructions. The efficiency of executing the instructions of a program is dependent on the skill and ingenuity of the programming group in writing the instructions.

A review of experience in selecting EDP programmers will be helpful in arriving at an answer to such questions as "What type of person will succeed as a programmer and what type of background is required? Is knowledge of the application or subject matter an essential ingredient?" Before answering these questions I would like to discuss with you some of my experiences in the Treasury Department, which I'm sure are similar to experiences elsewhere, both in and outside the Federal Government.

In the initial installation in the Treasury it was decided as a matter of policy to make every effort to recruit the necessary complement of programmers from those whose work was to be directly affected. This involved two large operating units comprising more than 755 people. In view of this policy, we were faced with the fact that not one in the group had acquired any previous experience in this field. In view of this, we decided to establish a program for screening and testing to select about 25 people to program the job. This job is known as the Payment and Reconciliation of Government Checks. We gave an aptitude test first. The test which was selected was especially designed to test the applicant's ability to think logically, rapidly, and creatively. 212 people applied for the aptitude test. The results of this test were not good when considered in terms of getting enough people to train for the job of programmer. Our goal was to come up with at least 40 to enter a six-weeks' training course. Only 15 obtained a passing grade and about 6 of these were in the lower third, or grade C. At that time we decided to test the "test." We selected a few more who were below passing in order to get a class of 21. All 21 of the final selectees were further screened through a supervisor's evaluation report to obtain knowledge of the applicant's willingness to work, ability to get along with others, skill in organizing his work, performance under pressure, and ability to complete his assignment, timely, with a minimum of supervision. In connection with the matter of willingness to work, attendance and punctuality were also considered. As the final step before training, the Committee which had been selected to supervise the installation interviewed each potential candidate. Based on the test, supervisors' evaluation reports, and personal interviews, 21 individuals were selected to attend a six-week training course in programming. Six of the 21 had obtained a grade in the aptitude test which was slightly below passing. During the training course we lost one who decided he was

not equipped for programming. 12 of the remaining 20 individuals made an acceptable grade for the course. At this point we had obtained 12 individuals out of 212 to be transferred to EDP programming group.

Another very important matter, which is inherent in the installation of EDP equipment, should be mentioned. Most employees will consider and ask questions about their future career. Principally they are concerned about their potential for earning better salaries and what the future holds in terms of advancement and job security. The Committee or head of the organization should be prepared to answer all such questions. It takes nearly a year of on-the-job training before a programmer can produce the type of work that is expected of him. Therefore, an organization has a sizeable investment in such an individual and good business seems to warrant consideration of two points, (1) salaries must be commensurate with abilities, keeping in mind also that there is a scarcity of trained programmers, and (2) competition with business concerns for this type of individual is extremely strong. This leads us to the conclusion that if a programmer stays on the job and produces effectively for two or three years, the organization's policy should be that promotion lines are open for better jobs throughout the organization and not be confined to the EDP function. There seems to be no doubt that the possibilities of attracting people with the qualifications for programming are better if the opportunities for advancement within the organization are present and the policy of management requires that all promotional lines are connected with the EDP group.

In order to fill the initial requirements of 25 we invited personnel whose work was <u>not</u> to be affected by the installation of EDP equipment to apply for the aptitude test. In this area the general grade structure and importance of work being done was somewhat higher. We, therefore, expected somewhat higher scores from our applicants than was obtained from the initial group. To a degree our expectations were warranted. We followed the same pattern that we had previously pursued except that no one below passing was accepted. This was because, of the 6 individuals of the first group who were originally selected with a sub-passing grade in the aptitude test, only one passed the training course. This individual's standing in the training course gradewise placed him about 9th or 10th in the group of 12. (It should be mentioned that this individual has become a very good programmer since selection about 4 years ago.) From the second group of 132 employees tested, we selected 20 for training. We obtained 11 from this group to supplement the original 12 to round out the programming group to 23 to start the job. In looking back on our experience in relation to questions, such as, "What type of person will succeed as a programmer and what type of background is required? Is knowledge of the application or subject matter an essential ingredient?", we find no positive

conclusion or definitions as to the exact ingredients which tend to make a good programmer. However, the especially designed aptitude test I referred to seems to be a good general indicator. But, it must be remembered that some of those with the highest grades did not make the best programmers. The last question can be answered definitely in this manner: A knowledge of the subject matter is not required as a basis for selecting programmers. In order that my statement is not misunderstood, I want to make it clear that I do not say that some subject knowledge will not be helpful to a programmer, but we know from experience that it is not a necessary attribute and should not under any circumstances be a prerequisite in the selection of programmers. To institute such a requirement will result in losing some very fine potential talent for programming.

More recently we have learned that some individuals who have passed the aptitude test designed for programming with excellent grades have been unable to pass certain other general tests designed to test their overall I.Q. and emotional stability. This is significant and should teach us that a generalized test of a person's I.Q. is not necessarily a good guide to bring out an applicant's ability to reason logically, rapidly, and creatively.

One final point which we have learned from experience is that we should refrain from trying to test the validity of a well-designed aptitude test. The odds are that a person who does not have the ability to demonstrate his expertness in logical thinking through such a test cannot become a good programmer within a reasonable time. It appears that mathematicians, systems analysts, and electric-accounting machine personnel fare pretty well in the aptitude test and apparently possess some of the important attributes of good programmers.

On-the-Job Training

Upon completion of the course in programming the most important phase of training begins. The selected personnel are indeed a long way from being programmers. Depending upon whether the EDP group already has an experienced programming staff, the methods for further training will take different courses. In cases where the organization already has a programming staff which has completed at least one program, experience teaches us that the new selectees probably can best be trained by experienced programmers with assistance from the systems analysts. More specifically, in such cases you can rely on the systems analyst and an experienced programmer to indoctrinate the new selectees in the logic and program required for a specific machine run. The new selectees then can be assigned to reprogram the application. Such experience can and should be rewarding. Under these conditions we can expect an acceleration in job training and possible benefits in terms of

improvement in the program which has been completed and is being used operationally.

An important point in programming efficiency is to set a goal for each program to be input or output limited. A programmer has no control over the time that is required to read data into and to write data out of a computer. This is true regardless of the form of the input and output. In other words, if the input and output of the computer is magnetic tape, the time required to read and write data from and on magnetic tape, as a minimum, is the time it requires to move the tape, and nothing that a programmer can do can change this timing limitation. However, most large-scale computers have the ability to read, process, and write data simultaneously. Hence, every programmer should establish as his goal the principle that his program should be designed in a manner that will enable the computer to process a given amount of data in no more time than it takes to read and write the same quantity of data. Maybe an example will help us to understand this principle. Assumption: Each 100 records of accounts receivable are grouped on the input tapes and the program has been designed to update an input master file by payments received and new sales to customers, and to produce a new output tape file representing an updated file of accounts receivable. Under these circumstances the computer is input limited. This is by reason of the fact that more records will be read from the input tapes than will be written on the output tapes. In such a case a programmer would design the program to read initially into the computer two groups of 100 input records from the master input file. Following this, the computer would then begin processing the first group. After completing the processing of the first group his program would be designed to perform the following functions simultaneously:

(a) Write Group A on the output tape.

(b) Process Group B within the computer.

(c) Read Group C from the input tapes.

If the program can be designed to accomplish step (b) in no more time than it takes to execute (c) the job will be input limited and in terms of time consumed on the computer it costs nothing to process and write the records. This illustrates the goal which every programmer should strive to meet, which is to process data within the time limits of reading and writing.

Problem Definition and Systems Logic

In cases where the first application is to be programmed by programmers with no previous experience, the approach should be to give the

programming staff an intensive indoctrination in the logic of the system and to define the problem in considerable detail. This approach assumes that the system has previously been designed. We cannot expect the staff of untrained programmers to go ahead without this assistance even though they may have considerable knowledge of the subject matter. Following the indoctrination it is necessary to keep in close touch with the programmers in order to see that their approach to the job is consistent with the logic of the system. The design of the system is by far the largest single part of the total job. Every effort should be made to provide the programming staff more training in systems work than is usually the case. The trend is toward this and should be rewarding to any organization.

Initially, the concept was to use the titles of EDP Systems Analyst, Programmer, and Coder. The EDP Systems Analyst was considered a step higher than a Programmer, who in turn was considered a step higher than a Coder. The distinction between the latter two was because the work of the Programmer was considered routine and required less experience and ability. This theory has practically disappeared now because we have learned that a Coder could easily misunderstand the objectives of the Programmer even though in some cases these objectives were clearly illustrated by detailed flow charts. Such a misunderstanding resulted in coding a program which was unsatisfactory and unworkable. Also, the development of automatic coding routines has reduced the work of coding. Hence, it is becoming increasingly clear that future developments will make it desirable for the programmers to receive more training in the techniques of Systems Analysis than has been the case to date.

Experience has taught us that one of the most important points to stress to a new staff is the necessity for good working habits. For example, many feel that the writing of a program which may be used only once a month and requires a small amount of machine time does not warrant as much attention to efficient time-saving techniques as a program which will be used daily. This appears to be a reasonable assumption on the surface. However, if we examine it a little closer we will find that in such cases we are starting the staff off with habits which may be practically impossible to change. Therefore, regardless of the relative importance of a program in terms of amount of machine time, the goal of every organization should be to establish the policy that every program will be as efficient as possible. This is true whether the program is a one-time job, rush job, or a daily job.

One final point on problem definition. We have learned that it is desirable for the systems analyst and untrained programmers to be in close contact with each other while the program is being written. Such an arrangement is necessary in order to establish a strong and continuous line of communication. This will transcend any other known method

for assuring management that all the requirements of a system are met by the program. It should be understood that a well-designed program must provide for any set of circumstances. It would be wrong for a programmer to assume that a certain set of circumstances could not happen. Programs will be prepared and written better when the programmer has the opportunity to pick the brain of the systems analyst during the preparation of the program.

Debugging and Systems Testing

When a program has been written it is not ready for operation by any stretch of the imagination. It must be debugged and systems checked. These terms are not synonymous. The term "debugging" is a process for finding and eliminating errors in the program as written. When this is accomplished there is no assurance that the program will accumplish all the logic that is required. You will note I used the word "required." A deficiency in the ability of a program to accomplish everything required will usually be found when it is systems tested. This may be accomplished by the preparation of test data covering every known condition existing in the application. These data should be developed in a manner which will test every instruction in the program. The use of these data in various combinations by the programmer during the desk-checking phase will be extremely helpful. Each routine (main, sub- and error) of a program should be "desk checked" prior to machine testing. It not only is considerably cheaper but also will result in identifying deficiencies which may be extremely difficult to locate after assembly and during machine testing.

Diagnostic Testing

After assembly of a written program the testing is usually done on the computer. This is accomplished by the preparation of input data which, while not necessarily large in quantity, will be representative of all input transactions. When the program fails to function there are several steps available to the programmer depending on the type of error. If the computer stops on a certain instruction or continues in a "loop" carrying out certain instructions continuously, a memory printout of these instructions should help to locate the error. The most troublesome and probably the most consistent type of error encountered during the initial machine testing is caused by instructions being modified during a step of processing when the programmer did not intend for them to be modified. We have learned that the programmer should list every instruction which he intends for the program to modify during processing. If this is done it will be possible to write a program which will "trace" or examine every instruction during processing and note those which are being modified. A comparison of the two lists will then indicate the error.

In summary, desk checking, test data, memory printouts, and test routines will enable the programmer to test and complete his program successfully.

Before leaving problem definition I would like to review with you some of the things we have learned about controls. First, let us examine the purpose of establishing controls over any process we wish to consider. The sole purpose is to assure us that the function has been accomplished in a manner which is consistent with our policy. This policy is translated to mean in most cases that the function has been accomplished without error. Controls over any process should be established in light of the consequences. Such consequences must include consideration of the degree of reliance on the output data by the user as well as the results in terms of costs for undetected errors. We have learned that electronic data processing equipment operates far more efficiently than any other type of equipment of which we have knowledge. The possibility of an undetected error being processed by an electronic computer is rather remote. If this is true we must ask the question "Why is it necessary to establish controls which may in some cases be extremely costly?"

The answer appears to be that it is not the rare undetected machine error which motivates the establishment of controls. It is the detection of programmers' errors, operators' errors, and bad handling and preparation of input data. Probably the most trouble which has been encountered to date and which dictates the necessity for establishing controls is with incomplete input data. The computer cannot detect this condition. Hence, it is desirable to establish controls for balancing the processing of input and output data. The danger lies in over-controlling. This may happen because it is relatively easy to program for all kinds of cross-checking and balancing. To illustrate this point to an extreme, it is possible to control and examine electronically every record during reading, each step of processing and writing, to determine that it has been processed internally in the manner required. This would involve a comparison of every digit and character of each record at each stage of processing. While this would not take much time for a single record, when multiplied by the number of records handled it would become probably more costly than processing. Therefore, it appears necessary to build into a system controls which will give reasonable assurance that the input data are correct and that such data as required have been written on output tapes.

If the controls over the function performed by the previous method were considered adequate, no additional controls should be required for electronic processing. In most cases controls can be reduced upon conversion to electronic processing techniques. This is for the reason that the possibilities of losing or misfiling records on magnetic tape are negligible in comparison with these possibilities under other procedures. One of the common types of effective control is to write as the last record

on each output tape, the number of records contained on this tape. When the output tape is read a record count can then be made and checked with the total appearing in the last record on the tape. This technique, together with automatic labelling of each tape to identify properly its contents with common information contained in the program before processing, has proven highly efficient.

Establishing a Time Table and Target Date

The development of a time table and reasonable target date depends on the amount of systems design included in the feasibility study. As previously indicated, if cost is an important factor then the feasibility study should have developed an overall plan, including a time table and a considerable amount of the system logic. Otherwise, it will be necessary to do these things before the EDP group is selected and ready to start programming the job. In any event a detailed systems plan must be developed before a realistic operating plan and time table can be established. When a good working knowledge has been obtained of the system logic, experience teaches us that it is possible to establish a plan and time table for installation.

Depending somewhat on the number and the complexity of the operations to be converted and depending on the size of the programming staff, target dates should be established to complete the initial application to coincide as near as possible with the scheduled delivery date for the equipment. In the initial applications it will probably require a team of programmers about six months to complete each program. In this connection, we find that teams of programmers consisting of two or three individuals will work out better for the development of each program. This arrangement will permit an interchange of ideas among the members of the team and also will provide the facility whereby each member's work can be reviewed by a person who is familiar with the problem. Another advantage is that such an arrangement provides a better basis for placing definite responsibilities for the completion of each program.

Delivery of equipment could be the controlling factor at the present time. This is because most of the major manufacturers are in the process of producing new "solid state" computing systems. These are commonly known as transistorized computers and have many things to recommend them in comparison with present computers which use hundreds of electronic tubes. In cases where delivery dates are a year or a year and a half away, it should be possible to establish a time table for performing all the preparatory work necessary to commence operations. However, as a general rule, it is wise to plan not to accept delivery of equipment until programs have been completely written and tested. In order to utilize economies as soon as possible at least one full shift of work

should be ready for the machine before delivery. This, of course, assumes that cost of operations is the prime consideration underlying the decision to install EDP. If this is not the case, then some special criteria will control the amount of work which should be ready at the time of delivery of equipment.

Generally speaking it takes about 4 to 6 months from the time the decision is made until the EDP group and programming staff can begin to produce any amount of acceptable work. This does not mean that work should be suspended on systems design, thinking ahead about conversion, site preparation, etc.

Conversion

Thinking ahead about conversion deserves serious consideration. Experience teaches us that one of the major problems encountered to date in the installation of EDP equipment can be traced to the fact that this phase has not been carefully planned. This is especially true in cases where the new system requires the establishment of master files with records which contain historical data. If possible, in designing the system consideration should be given to providing for the new system to build up its own historical data even though the application in its entirety cannot be converted initially.

Conversion may be much more costly than was originally contemplated. In many cases conversion problems result in converting punch card files to tape because they contain many errors. These can be classified as (1) off punching, (2) double punched, (3) blank columns, and (4) inaccurate control punches. All such errors have to be corrected before a file can be converted successfully to magnetic tape.

In view of experience gained in the problems encountered during conversion, it appears desirable to organize a team or task force in the early stages of installation with assigned responsibility for determining that each step of conversion will be completed on time. There have been cases where inadequate planning in this area has delayed complete conversion for over a year.

One final point on conversion, clerical procedures and reorganization of divisions, branches, and sections may require attention. This usually is the result of changes required in input information and because output may be considerably different in form and content than was the case under previous procedures.

Site Preparation

Site preparation will usually require 6 to 8 months from the time

the decision is made to install EDP equipment. This includes the time required for architectural drawings, advertising for bids, and actual work by the contractor. In this connection an important consideration is the type and amount of air conditioning which should be provided to keep the equipment in operation. While manufacturers are able to specify with accuracy the air conditioning requirements for the complement of equipment to be installed, they do not usually assume or point out that any air conditioning system, of the magnitude usually required, may fail to operate occasionally for any one of a number of reasons. Experience has taught us that during the planning phase engineers who are charged with the responsibility for designing the detailed specifications for the air conditioning system often are not advised of the necessity for providing standby equipment in case of failure. If the application to be converted to electronic processes is a critical one, then management has the responsibility to advise the architects and engineers of their requirements for continuous operation. Maybe, if we consider the consequences, in the case of a large payroll operation, if a condensor evaporator motor in a thirty-ton air conditioner system burns out let's say during a time of nationwide strike in this industry, it will be helpful to focus attention on the importance I attach to providing standby air conditioning equipment during site preparation. The additional cost for this type of insurance is relatively small in comparison with the total cost for converting to electronic techniques.

It is at this stage that serious consideration has to be given to future needs in terms of expansion and possible change to newer and better equipment in the next few years. If possible the computing area itself should be sufficient to bring in additional equipment to take care of increased workload. In connection with future developments and possible changes in the complete system within a few years, consideration should be given to the potential for adjacent space which could be easily converted for the use of new equipment in a few years. During this time this space should be allocated to operations which could be easily relocated. The reason behind this type of planning is to anticipate the possibility of converting to a new and advanced system some time in the future, which will probably require parallel operations for a few months.

Parallel Operations

Wherever possible and practical, provision should be made to conduct parallel operations during the initial stages of conversion. It is not necessary to conduct these parallel operations with current data. It may be more practical to plan in advance to accumulate and store current data which can be used by the EDP system at a later date. It must be remembered that the EDP system which has been completely tested and debugged, and is supposedly error free, is the result of the work of the systems

designer and programmers. Often a particular set of circumstances, which may under actual conditions occur frequently, is overlooked in the design of the system. Hence, parallel operations with actual data will provide a sound basis for testing the new system under actual operating conditions. Also it will be helpful in building the confidence of all concerned in the reliability of the new system.

A word of caution: Do not continue parallel operations after they have served their purpose because you are now finally at the stage where savings can begin to be realized.

Use of Service Bureaus

Before I conclude I would like to discuss with you the matter of utilizing service bureaus. For many years service bureau facilities have been available to users of tabulating equipment. I am sure that there are a great number of cases where the use of these facilities has proved beneficial to Government agencies in the past, and it will probably be true in the future in the case of EDP equipment. However, I do not believe that it would be economical for an agency to continue to use service bureau facilities in lieu of establishing agency systems. This does not mean to imply that an agency should plan to install EDP equipment for a one-time job.

Much thought and discussion is currently being given to the feasibility of establishing Government owned and operated EDP service bureaus. As in any matter of this magnitude, there are divergent points of view. These differences of opinion are beneficial because they result in examining the issues more closely and help to focus attention on factors which otherwise might be less than fully considered.

Aside from the feasibility of having functions of various Government agencies performed by Government operated service bureaus, there are for consideration the benefits which might accrue to the Government as a whole under such an arrangement which would have the facility to carry on experimentation and training in the field of EDP systems.

SCHEDULING OF PRE-INSTALLATION ACTIVITIES

V. J. Fogarty

Vincent J. Fogarty is engaged in planning for installation of an information retrieval system for use with the EDP inventory control installation at the Ships Parts Control Center of the Department of the Navy, Mechanicsburg, Pa. He was Chairman of the group which designed this EDP system, has been working in data processing activities for about 20 years, and is active in professional associations in his field.

The opportunity to present this session of the Graduate School's seminar on automatic data processing is a pleasant assignment and something to which I've been looking forward.

I like to talk about EDPM because the possibilities it offers are so tremendous. This is really a big subject and it reminds me of the familiar fable about the four blind men who came upon an elephant and, after investigating the strange creature, tried to describe it. You'll recall that the first man seized on one of the beast's legs and reported that the elephant resembled a tree trunk.

Another of them found himself up against the creature's side and described an elephant as much like a rough and weatherbeaten wall. A third chanced to grasp the animal's trunk--to him, the creature resembled a heavy hawser, or a great snake, or serpent of some kind.

The fourth man grasped, I believe, the elephant's ear and discovered that the animal was very like a bird with heavy, flexible wings. Or it may be--I'm not too sure of my recollection here--that the fourth took hold of another part of the creature and received a distinctly different impression. It doesn't really matter. The point is clear enough--that our reactions to a thing seem to depend on what part of its anatomy we grasp.

Now I'm not going to try to describe in detail the whole elephant, but rather just that part of our EDP program concerned with pre-installation planning after it was decided that we would install a machine. Then we can discuss very briefly what has been going on since our first machine was installed in 1956.

Before we get into a detailed discussion of our EDP program I believe we should spend a few minutes discussing the Ships Parts Control Center and what its mission is.

The Ships Parts Control Center--SPCC--is the Navy's Supply-Demand Control Point, or inventory manager, for ships' repair parts. Its mission is to insure a proper balance between the supply of and demand for these parts. The SPCC is assigned the inventory control responsibility for such items as: (1) equipments and repair parts for Internal Combustion Engines; (2) Gyrocompass, Main Propulsion, and Navigation equipment; and (3) many other electrical, hull, and machinery equipments or components.

SPCC is the catch-all Supply-Demand Control Point. If the components or equipments installed aboard ship are not pure electronics, pure ordnance, pure aviation, peculiar to submarines, or commonly used everyday general stores items, the repair parts support responsibility is usually assigned to the SPCC.

Technically, SPCC comes under the control of the Bureau of Ships and under the management control of the Bureau of Supplies and Accounts. It maintains a centralized inventory control on 135,000 items. At one time the SPCC maintained control of more than 300,000 items in store, mainly due to the various types of equipment it is required to support.

But many of these items were added to SPCC's inventory list during wartime and some were based on the recommendations of manufacturers that certain repair parts should be carried in the supply system. A lot of this material did not "move," and has been disposed of. At the present time, the SPCC controls an inventory valued at approximately $450 million.

It has been estimated that the SPCC could be called upon to supply, either from stock or by purchase, a range of from one to three million items of repair parts. Because it is impractical to stock all of these items in the supply system, centralized inventory control is maintained over only 135,000 items.

Our investigation for the possible installation of an EDPM started in 1952, at least in the dreaming stage, and installation of card-to-tape equipment was made in May 1956, followed by the central processing unit in August 1956.

I have prepared a pre-installation planning chart which I will use as a guide for our discussion this morning. This chart covers a period of 15 months prior to installation of the machine. It is considered that about this much time is required in order to get ready for a machine. There's a lot to be done during this 15 months and it will be helpful for you to have an outline to follow.

Organization - 15 Months

The first action required is to insure satisfactory installation of the data processing equipment.

Executive Committee

The extensive changes in organizational responsibility, personnel, systems, and methods, and the large potential for dollar and personnel savings that result from EDPM use require that top management direct, coordinate, and approve the EDPM program. To insure this direction, it is recommended that an EDP Executive Committee be established and charged with the full responsibility for progress in pre-installation planning and for necessary management decisions affecting policy, procedures, personnel, etc. This committee will evaluate all phases of the operations for top management.

Project Director

Reports to the Executive Committee and is responsible for supervision of all details of the installation program including coordination, assignment, scheduling, and follow-up.

Analysis and Programming Staff

This staff will be assigned to specific duties by the Project Director: It will (1) make studies in areas selected; (2) design procedures for data handling; (3) write programs; (4) prepare test data for the various machine runs: (5) advise the Project Director of its progress; and (6) report on all phases requiring management decision.

At Mechanicsburg we had no formal directive for initiating the original feasibility study. The EDPM study was initiated in 1952 by key personnel of the Machine Records Division. Our initial report was approved by the Commanding Officer and forwarded to the Bureau of Supplies and Accounts by letter dated 6 April 1954. The Bureau approved the SPCC recommendations for installation of a machine and placed a Letter of Intent with IBM for delivery of a 702. We never did install the 702, however, because IBM made only a small number of these machines. If my memory serves me correctly they built only 18 of the 702 machines. We installed the first 705 in the Navy.

Our Executive Committee was established in June 1954. Members were selected based upon their intimate knowledge of specific areas for application of EDPM techniques. The committee was comprised of subject matter specialists and top level supervisory personnel who would ultimately be responsible for implementing committee recommendations. Commander H. P.

Mills, Machine Records Division Director, was selected as Chairman by the Commanding Officer, in view of his previous data processing experience and the part he played in developing the initial feasibility study.

Administration--Education

Your Project Director must establish administrative plans and objectives for the groups involved. He must set up lines of communications and reporting procedures and provide for follow-up on problems and progress. In order to assist each of the administrative groups in the proper evaluation and execution of its respective functions, arrangements must be made to provide sufficient data processing machine training. This education should follow the general pattern indicated.

1. Executive Committee

Each member of this committee should attend a Data Processing Executive Class. This training should also be supplemented by orientation seminars and demonstration visits where possible.

2. Project Director

Should attend Data Processing Programming Class. He should also be included in any of the seminars or demonstrations conducted for the Executive Committee.

3. Analysis and Programming Staff

All members of this staff should have a thorough knowledge of programming. Each member of the group should complete a programming class.

Scheduling

The Project Director, at this time, should discuss with the Executive Committee the projected delivery schedule. He should emphasize the time required and the work to be accomplished during the pre-installation period and insure that both the customer and the manufacturer will be ready for machine installation on the date of scheduled delivery.

I can assure that if you are able to accomplish the objectives set forth, so far, 15 months prior to installation of your machine--you are in excellent shape.

Keep Up The Good Work!

Throughout the course of this 15 months period your Project Director is going to be "involved" in many, many things and you will need someone who won't lose sight of the "big picture." Speaking of "getting involved" in things reminds me of the announcement of a professor's new automatic computer and his wife's new baby which appeared almost simultaneously in the local newspaper. Upon being congratulated on "This proud event in the family," the professor naturally thought first of the achievement that had cost him the greater effort. "Thank you," he replied modestly, "but I couldn't have done it without the help of two graduate students."

You may not agree with me when I say that the professor lost sight of the big picture. So, I say, be very discriminating and selective in deciding who your Project Director shall be.

Objectives--14 Months

Objectives

At this time the Executive Committee should be preparing a broad definition of the objectives of the program, the selection of applications and their priority, and also the target dates for completion of various levels of activity. You should consider the following points in application selection:

1. Desire for rapid cut-over operations.

2. Desire for early savings.

3. Possibility of centralization.

At Mechanicsburg, inventory control (redistribution, reallocation, and procurement) and the maintenance and dissemination of technical engineering records were the starting applications.

We could foresee the day when many significant improvements could be achieved if high speed data processing at SPCC became a reality. For example, at the time we submitted our request to the Bureau for installation of an EDPM, the SPCC was receiving quarterly stock status reports on each item of ships parts carried at our reporting activities. These items were reported on whether any transactions had occurred or not. The information was in summary form and the volume of cards was so great that more than a month went by before our EAM machines produced the consolidated report for review by our Stock Control Analysts.

We knew that a great many things were happening to our material, but we did not have the means always to take definite and timely action. That is one of the primary reasons why we wanted to obtain an Electronic Data Processing Machine.

We wanted to change our reporting system from the quarterly, summary type of field reporting to a transaction reporting system. By the use of high speed magnetic tape machines, we would be able to process our reports rapidly and take timely action. Our plan for changing the reporting system was approved and implemented in October 1956.

With transaction reporting and the use of management by exception techniques, all items are reviewed periodically, but only those items requiring action are printed out for the Stock Control Division. In doing this, the machine makes many decisions. This permits Stock Control personnel to devote more time to those items requiring attention. Also, all the known facts are associated with the item. This was not possible with the old punch card system. We have combined into our Master Tape records all of the information previously contained in ten different punch card files. All of this information is now consolidated on one piece of paper--the CSSR.

The items which need attention are processed by the machine with resulting automatic actions proposed to the Stock Control Analyst.

Another important factor is that the machine is always consistent, but many times people working the very same instructions come up with conflicting answers. The machine processes an item the same way <u>all the time</u> because we have instructed it to do so.

<u>Maintenance of Catalog and Technical Record Files</u>

Another really big application for the EDPM was automatic record maintenance.

Our Technical Records files (about 10 million cards) were converted to tape in September 1956 and have been maintained on a monthly basis since that time. Maintenance every month was not feasible economically with conventional EAM equipment and, under the old system, approximately six months were required to complete a full processing cycle. The 705 has made it possible to maintain the file on a current basis.

I might say that we were pretty successful in installing our machine--but I really think it was the result of establishing our objectives and planning to meet these objectives. We stressed the importance of the organization behind the machine. The ability of those in charge to keep

their feet on the ground and not float away on cloud 9 was also a valuable attribute.

General Flow Chart

The Project Director and the Analysis and Programming Staff should prepare a general flow chart of the objectives prescribed by the Executive Committee. Emphasis should be placed on responsibility of the Analysis and Programming Staff and the importance of a schedule for project completion.

The general flow charts prepared at the time of the feasibility study could be updated for this purpose, and once completed, should be reviewed by the Project Director and the Executive Committee. Approval should be obtained from all responsible executives. Flow charts show major machine operations and are not intended to display the machine operation in detail.

A narrative explanation should accompany the flow charts to explain each major operation in detail. The narrative must be explicit and correct, as machine operation will be tailored from it. Consolidation of present files should be noted at this point. The programmer may elect to make a further consolidation or breakout of files to facilitate machine operation, but in general the decisions made now will hold true.

Policy Changes

Policy changes resulting from the general flow chart, or changes that affect personnel, procedures, and public relations should all be discussed with appropriate personnel.

See to it that necessary steps are taken to initiate these changes as they occur.

Personnel

The Project Director should work together with the Personnel Department on manpower requirements. He should give consideration to the type of personnel, whether they be:

a. Supervisory level--could come from the Analysis and Programming Staff. If they don't, they should attend programming school.

b. Tape Librarian--could come from Analysis and Programming Staff. If they don't, they should attend programming school.

c. Console Operators--should have console and machine operation training. Should be able to read and understand a program.

d. Machine Operators (punched card and EDP)--should be given normal training for EAM equipment but Computer Operators should be given training in EDP operation and card and tape handling. We have even gone so far as to have our GS-5 Computer Operators go to programming school after they have been on the job about six months. They like to have the opportunity and we feel it pays big dividends.

Personnel Selection

We had a real problem in the personnel area in staffing the positions of responsibility. We were very careful in selecting our EDP Staff. Our selection panels spent a great deal of time and effort reviewing the past experience and the work records of the candidates for our positions. For the persons to be interviewed we prepared our questions in advance so as to be certain to cover everything on which the panel wanted to obtain additional information.

It was hard for our employees to meet the minimum Civil Service requirements for the Computer Programmer positions. In order partially to overcome this, we entered into a promotion and training agreement with the third regional office of the Civil Service Commission so that our employees might qualify for some of these jobs. For example, if an employee sucessfully completed the training program, he would qualify as a GS-9 Computer Programmer, obtaining the required specialized experience necessary in only nine months of training. In order to help make sure that the employees given the training would successfully complete the course, 89 applicants were tested for the first 8 programmer positions. We interviewed the top 22 applicants, and selected those who were considered as having the best overall knowledge of the Ships Parts Supply System, or at least had a thorough knowledge or familiarity with one phase or area of the system--such as Stock Control, Technical Records, Catalog, etc. By carefully selecting our candidates, we were confident that they would successfully complete the training program, and would, after they had finished the training program, be good employees for the new organization. We hoped we would not be just a training ground for industry. By being up in the sticks at Mechanicsburg, and by selecting candidates having some ground roots to the area, we hoped that we would not be faced with a big turnover problem.

It's been our practice in our interviews to select applicants who have established permanent residence in the area--people with a defined motive for applying for data processing work, and who display the proper attitude towards a career in Federal Service.

Restudy Selected Applications--13 Months

Restudy Selected Applications

The Project Director and the Analysis and Programming Staff should make a detailed survey of the selected applications. This survey should emphasize the following points:

1. Observe objectives and results rather than the techniques used.

2. Take note of "cause and effect" areas relating to results in the base applications under study.

3. Pay special attention to exceptions. Can be very troublesome, must get decision rapidly.

4. Note existing due-in and due-out schedules. What is effect of EDP?

5. Analyze present manpower requirements--increases as well as decreases.

6. Consider all costs of present operations.

7. Rely on the normal survey techniques in the examination of:

    a. Source Documents.
    b. Files.
    c. Reports.
    d. Procedures.

Orientation

The Project Director should cooperate with operating officials in the preparation and conduct of orientation programs for the personnel in affected areas.

The personnel situation that exists when an EDPM is to be installed

can be a very touchy one and I'd like to tell you how we coped with it, but I believe it appropriate to discuss the entire personnel situation rather than just that portion concerning the EDPM operation. The Command at SPCC displayed a real concern for employee morale. The Command placed sincere emphasis on the need to keep employees informed of the progress of EDP planning. Naturally, all employees were wondering how the machine would affect their jobs and I believe a great deal of our success can be attributed to the fact that the employees were kept informed. After the machine was installed, all employees of the Ships Parts Control Center were extended an invitation to visit the Data Processing Center and see the 705 machine in operation. The personnel were scheduled in groups of 35 at about 20 minute intervals. The visit began in the EDP "viewing room" where each piece of equipment was discussed. Interesting statistics and the speed of operations were highlighted. Next was a tour through the EDPM room itself.

Throughout the 20 minute visit, the underlying theme was to convey to these people that they would, either directly or indirectly, have some part in preparing information that would become an input to the machine or that they would have some part in working with output information that had been processed by the machine. It was made clear that the machine could quickly perform their routine clerical operations, permitting them more time for the important aspects requiring evaluation or decisions.

The effort we put forth in keeping the employees apprised of the EDPM program paid big dividends and when it came time to reduce personnel, it was done without anyone getting hurt. The EAM tabulating equipment operators positions were the ones most affected and most of these people found jobs as clerks or worked into the EDPM program. A number of employees transferred to another Naval installation at Mechanicsburg and still others obtained jobs in industry. The significant thing about all this is that an informed group of people cooperated fully in this endeavor. Everyone seemed to consider the EDPM operation as another forward step of progress.

We had other training to do also. We had to indoctrinate key management personnel. To acquaint top level supervisors with the operations of high speed data processing equipment, military and key civilian personnel were progressively scheduled to attend a one week Executive Course conducted by IBM. We went "all out" on this type of training and even had several classes conducted right at Mechanicsburg. By the time the 705 was installed over 150 people had been afforded this type of training. You might even say we "brain washed" top supervisory personnel because we wanted their support.

Site Planning--12 Months

Space Requirements

The Project Director and Executive Committee must make the initial determination as to the best location for the machine. Usually there are only one or two logical places to locate the machine. Many of the EDP machines require a controlled climate in which to operate, and thus a building which is already air conditioned will result in cheaper site costs.

The Project Director should request early assistance from Public Works Department personnel. In relation to size, consider the components on order, the tape file storage requirements, and the needs of the customer engineering group.

Prepare a preliminary layout. Consider again the points mentioned above as well as the requirements in the physical planning pre-installation manual provided by the EDP manufacturer. Take definite steps toward selection of a contractor.

Power Requirements

The Project Director, Physical Planning Engineer, and the Building Engineers should review the power requirements as set forth in the manufacturer's physical planning pre-installation manual. Consider all phases of power requirements including the possibility of both data-processing machines and punched-card equipment.

Air Conditioning

It may be wise to use an outside consultant for A/C advice. Make sure that all responsible persons are in complete agreement as to your requirements. Stress need for control over conditions required by the physical planning pre-installation manual.

Construction Plans

The Project Director and Physical Planning Engineer should insure that definite steps will be taken towards completion of the necessary construction work, selection of a contractor, and acquisition of necessary equipment. Make arrangements for a meeting to review progress of construction planning.

This Project Director is a busy boy, isn't he? Believe it or not, the site construction problem proved to be very difficult and time consuming for us. IBM suggested that we start real early on site construction plans

and we are most thankful that we heeded their advice. Our records show that as early as December 1954, some 19 months before our machine was to arrive, we reported to the Bureau of Supplies and Accounts:

> "Tentative plans for the physical installation of the electronic data processing machine have been developed. The Public Works Officer of the Naval Supply Depot, Mechanicsburg, is submitting a letter of request to the Bureau for approval of an architectural and engineering contract prior to receipt of funds from the Shore Station Development Project for the installation. It is believed that such a contract would be warranted because of the technicality of the air-conditioning and humidity controls necessary for this type of equipment."

Block Diagramming--11 Months

Completion of Restudy

The Project Director and the Analysis and Planning Staff should complete the restudy of application areas during this month. Since the block diagramming is to be started in this period, manpower of the Analysis and Programming Staff must be diverted to this work.

Block Diagramming

Block Diagrams are detailed machine operation charts. These charts are developed by the programmer and require a very detailed knowledge of the machine. The actual coding of the instructions will be done directly from these charts.

The Analysis and Programming Staff should begin preparing a block diagram for each data processing machine run. Since this is a graphic representation of the logical development of the problem, consideration should be given to the following points:

1. Control of data before the machine pass.
2. Type of input.
3. Programmed controls.
4. File Maintenance.
5. Program checking and error-correction routines.
6. Type of output.
7. Control of data after data processing machine pass.

Block Diagram Review

Errors in the problem definition or logical solution can be costly and time consuming to correct. It is important that a review of the block diagrams be conducted as they are completed.

The chart I have prepared suggests that block diagramming be completed over a period of six months, but it may actually take a little longer because coding and block diagramming are somewhat closely related. Estimated 75% of job is flow charting and block diagramming; 25% coding.

Our historical records tell us that we were doing our block diagramming about 15 months prior to installation of the machine for our stock control application. The following is a quotation from a report we submitted to our management Bureau in May 1955:

> "During the months of March and April, preparation of detailed block diagrams for the stock control problem was undertaken. The stock control application has been divided into three segments as follows:
>
> 1. Daily processing for updating and Perpetual Inventory Records.
>
> 2. Preparation of shipment orders, redistributions, reallocations, procurements, etc.
>
> 3. Preparation of inventory analysis, budget statistics, and processing permanent changes to the perpetual records caused by stock record change notice actions.
>
> It is considered that progress in this area has been satisfactory. The Stock Control Division has provided detailed criteria for use by the machine in making procurement recommendations, redistributions, etc., for the fast moving items. Detailed criteria for items other than the fast movers are presently being developed by the Stock Control Division. It is expected that these guidelines will be completely finalized within the next three to four weeks. The actual writing of program instructions for preparation of inventory analysis reports is already in process."

Programming--10 Months

Continue Block Diagram

The Project Director should insure that block diagramming continues on schedule.

Continue Block Diagram Review

Check on progress of the review of block diagramming. See that the review is keeping pace with the block diagram preparation.

Programming

The Project Director should assist the Analysis and Programming Staff in writing programs from the block diagrams. Consider the following points in the organization of this work:

1. Program teams. Number of programmers to a team, area to be handled by a team, number of program steps within the area, and scheduled completion dates. Teams work well especially when they write their first program. Can help each other.

2. Utility routines. What routines are required? Sorting routines, print memory, End-of-File, etc. What is available from manufacturer of equipment? Many programs are available from IBM.

3. Check Point and Restart procedures.

Shift Usage

A meeting with the customer engineering manager should be set up to review plans for customer engineering coverage. Establish the number of shifts required and the components to be used on each shift. Review the extra shift policy. Review schedules for diagnostic routines and bias checks.

Program Testing--9 Months

Block Diagramming

The Project Director should ascertain that all block diagramming operations continue during this period and that they will be completed about five months before installation of the machine.

Review of Block Diagramming

During the month the review of the block diagrams should continue until completed. Follow closely to see if additional manpower will be required to complete this work on schedule.

Continue Coding

The Project Director should follow up on coding progress. As block diagramming is completed, more effort should be shifted to the programming phases.

Program Testing--Anxious to Test a Program

Prior to the arrival of the machine, testing at one of the test centers should take place. This is necessary in order to have the programs checked before arrival of the machine.

An early test session is recommended to familiarize the programmers with the machine. It is also a boost to morale as many of the programmers have never seen a machine. The first test session also serves to tell whether the programmers are on the right track in their machine logic. The first test session should be held as soon as there is sufficient programming completed to make the time worthwhile.

Test schedules for the rest of the test time available should be worked out well in advance. The last test session should be completed about one month before machine installation.

Only completed programs should be taken to the test sessions. These programs should have been well desk checked and any revisions should be made before the test session. There is not time to make any major changes in a program while at the test center.

The project Director should consider special requirements for the test sessions such as security clearance, visits by executives during testing operations, and programs that exceed the capacity of the testing machines.

Again quoting from our historical records I can tell you that:

> "On 7 and 8 November 1955, the Ships Parts Control Center completed the first check out period on the IBM Type 702 Electronic Data Processing machine at the IBM Data Processing Center, New York City. The program to build perpetual inventory records for the stock control application was thoroughly checked. This problem required eleven hundred forty (1140) program steps and was processed without error the very first

time SPCC representatives were given machine time. This Command is proud of the personnel who contributed to this remarkable feat and is pleased to be able to submit such information to the Bureau of Supplies and Accounts. Approximately seven thousand (7,000) program steps have been written for the inventory control application. It has been estimated that this entire area of work will require from 10,000 to 12,000 steps. It is planned to test this program at the Data Processing Center in New York in March."

We later reported that "during the months of April and May 1956, visits were made to the IBM Data Processing Center, New York City, for program testing purposes. Twelve (12) machine runs involving 18,445 program instructions were successfully tested."

Special Requirements

The Project Director should check with the Analysis and Programming Staff on the details of any special requirements and/or devices. Watch for the necessity of delivery schedule tie-in for punched-card equipment. I believe I mentioned that we received our card to tape equipment in May 1956. We got this equipment about four months prior to the central processing unit so that we could get our cards converted to magnetic tape well in advance of installation of the main frame.

It became apparent to us in November 1955 we had a need for special requirements. The following is a quotation from a report to our management Bureau:

"The recent announcement by the IBM Corporation of the availability, after October 1956, of 20,000 additional positions of magnetic core storage on the 705 machine was received with extreme interest. In the inventory control area the use of a central processing unit with 40,000 positions of magnetic core storage would eliminate the necessity of two machine runs each week and four machine runs each quarter. This is the equivalent of seven hours machine operating time each week and would provide additional machine time for the allowance list program, as well as for the conversion of additional programs to EDPM processing in the future. Because of the long lead time involved, it is recommended that reference (b) be amended to include 20,000 additional positions of magnetic core storage for installation as soon as possible. The increased rental amounts to $2500.00 per month. While the SPCC is not certain at this time, it is nevertheless believed that the additional storage will be sufficient to obviate the need for the magnetic drum. However, it is felt that the best interests of the Navy can be served by having the magnetic

drum also remain on order until the requirements of the Simplified Allowance List Program are fully defined. In this regard, the Ships Parts Control Center recently completed a test on the preparation of thirty of the new Allowance Parts Lists for the Bureau of Ships to further develop this program."

Time and Cost Analysis--8 Months

Time and Cost Analysis

The Project Director and the Analysis and Programming Staff should establish time and cost factors to insure that the equipment will do the job and will achieve the results you desire. These factors are the major elements in the justification of the system. Consideration should be given to the components on order; both data processing machines and punched-card equipment, as well as to the cost figures discussed during the survey. It's most important to keep these data available for later management review.

I'm sure you all know where the management review people come from. They are people who would have been poets, historians, or biographers, etc., if they could: they have tried their talents at one or the other, and have failed; therefore, they become reviewers. Or as Disraili has said; "It's much easier to be critical than to be correct."

Seriously, you should review the cost and time figures with the Executive Committee. Obtain approval of all factors. Insure that action is taken when required; e.g. additional components to be ordered or released.

Conversion Requirements

The Project Director should establish schedules for conversion operations. The following points should be considered to see that steps are taken to meet these requirements:

1. Procedures and controls.

2. Component requirements.

3. Tape usage.

4. Customer Engineering coverage.

5. Power, space, and air conditioning.

6. Delivery schedules.

There are Two Main Methods in Use for Conversion:

1. "One Time" conversion will gather all the data needed to start the operation at once. It will all be processed through the conversion programs at the same time. Once that has been completed, regular operations will start. This has the advantage of being very straightforward and eliminates the operation of two jobs at once. However, if the volume is very great, this may not be possible. Before the conversion is finished, a large backlog of transactions may build up and would be difficult to process.

2. Phased conversion, which builds up the data needed, a section at a time. This, of course, means that the data already gathered and placed on the EDP must be maintained through the use of the regular operating programs. The obvious advantage here is that the conversion workload can be spread over a period of time. The disadvantage is that you must run two jobs at once in the EDP, conversion and operations.

Each conversion problem will have to be considered in its own light to determine which is the best method to be used. A schedule should be set up for these operations and all personnel affected should be notified of the part they will play. It should be emphasized that a time cushion should be built into the conversion schedule. Experiences of past conversions have shown that, regardless of the planning done, unforeseen circumstances will arise that will slow the schedule down. If a proper cushion has been built into the schedule, these circumstances can be handled without delay to the operation as a whole.

At Mechanicsburg several months prior to installation of the "main frame," we arranged to receive the basic card-to-tape peripheral machines in order to commence an early conversion of our 34,000,000 punched cards. We recognized that conversion of millions of cards to magnetic tape was a lengthy process and that during the conversion period normal SPCC business operations had to continue.

To cope with this problem, we introduced parallel operations whereby card decks were converted to tape immediately following normal EAM operating schedules. The cards were then returned to the EAM room to continue our business operations. As daily changes were made to the original card files, duplicate transactions were reproduced and held for updating the tape records upon installation of the computer. This operation was employed throughout the preliminary stages of our "getting ready" for the computer.

<u>Continue Coding</u>

Follow up on progress of coding to see that it is proceeding as

scheduled. The Project Director should advise the Executive Committee of any deficiencies and suggest corrective measures to maintain the schedule.

Accessories, Supplies, and Equipment--7 Months

Accessories, Supplies, and Furnishings

The Project Director should determine his requirements for miscellaneous equipment and consider the needs for:

1. Forms, internal and external.

2. Card requirements.

3. Magnetic Tape Requirements.

4. Tape File Requirements.

5. Furnishings, etc.

Continue Coding

The Analysis and Programming Staff continue coding during this period.

Program Review

During this month the Project Director and the Analysis and Programming Staff should initiate a review phase of the programming effort. The programming effort should be about 50% complete. If not, more effort should be channelled to programming. Pay particular attention to clerical errors, errors of interpretation of the block diagrams, and completion of the End of File, error correction, and automatic Restart routine.

Site Planning Follow Up

Arrange for a meeting with the Public Works Engineers. See that reports of progress are made by all interested parties regarding letting of bids, selection of contractors, the ordering of necessary materials and supplies, delivery dates, and projected completion dates. It would be wise to review the floor plans in light of any changes made in components on order.

This is an area where a generous cushion should be added to be on the safe side. Having a room stand idle for a month or so is desirable rather than the other alternative--not having the room ready when the machine arrives. The odds are that at best the room will be ready only at the

last minute, in spite of your efforts to have it ready a little ahead of time.

Machine Loading--6 Months

Machine Load and Work Scheduling

The Project Director should establish time schedules and priorities on data handling. He should schedule requirements for:

1. Main Frame
2. Auxiliary Operations
   Card to Tape
   Printer
   Tape to Card
3. Pre-processing Schedules
4. Due-in and due-out requirements

Desk Checking

Insure that all programs be given a thorough desk check. Experience has shown that about 90 percent of program errors are clerical and that the vast majority of such errors can be eliminated by proper desk-check procedures.

Test Data Preparation

Assist the Analysis and Programming Staff in the selection and creation of test data. Care should be taken to insure that the data used will test all conditions written in the program. The data should not be in large batches. This consumes machine time unnecessarily. Explain that a small quantity of carefully prepared test data will suffice.

Complete Program Review

The Analysis and Programming Staff should complete the program review phase during this period. In order to insure effective use of the program testing time, programs must be completely checked for elimination of the so-called "clerical" errors. A great deal of lost time during the program testing session is caused by machine stoppage due to program errors that should have been detected by a careful review of the program.

Finalize Plans for Conversion--5 Months

Conversion

During this month final plans for conversion of records to tape should be reviewed and approved by the Executive Committee. Operators and

Programmers should all be apprised of their responsibilities for the conversion effort.

Program Testing

The Project Director should maintain records of progress on program testing. He should note production statistics and compare them with the test center average. Keep track of types of errors encountered. Be especially watchful for errors in logic. Keep a current projection on the completion date of program testing. Insure that sufficient programs will be checked out so that productive work may start as soon as the machine is installed. Advise the manufacturer's representative immediately if additional time will be required, or if scheduled testing sessions will not be required.

Review of Power and Air Conditioning

The Project Director should arrange a meeting with the Public Works Department Engineers to finalize plans on the machine room layout. Review delivery schedules of material on order. See that all persons involved are in agreement on completion dates for all phases of construction.

Program Revisions--4-3 Months

Operator Training

The Project Director should arrange for training on data processing machine operations. Consider the following possibilities:

1. Operator training during testing sessions.

2. Console training at customer's office.

Site Construction

Actual construction of the machine room should start this period. The Project Director and the Executive Committee should review pre-installation activities, and discuss progress to date and the work remaining. Review, with the Executive Committee, the plans for installation and the results to be accomplished within a specific period of time. Assure yourself that all members of the Executive Committee and top management are in agreement with facts presented and objectives to be reached.

Follow-up Accessories

The Project Director should follow-up on the delivery schedules of supplies, accessories, and furnishings. Review delivery schedules on tapes,

punched-card equipment tie-ins:

Tape Cabinets

Reels of Tape

Ribbons

New Forms

Final Program Testing--2 Months

Assignment

The Project Director should assign personnel to specific duties and responsibilities. He should insure that all persons involved are aware of details of their jobs. Discuss shift make-up if required. Review plans for:

1. Operation of the installation.

2. Future work loads.

See that the strength of the Analysis and Programming Staff is spread over these areas.

Program Testing

Plan to complete check-out of all programs necessary for use on installation date. By the end of this month all details on machine procedures should be completed, tested, and ready to run.

Program Revision

Since the program testing sessions are due for completion during this month, the program revision phase should be completed. It may be necessary to "put a freeze on" changes to programs until after the machine is installed and operating.

Physical Requirements

Insure that you are able to install the machine when it arrives. Arrange for testing of air conditioning equipment two weeks prior to machine arrival. Final balancing of the system may be done after customer engineering has turned on the power.

Final Details--1 Month

Review of Final Details

The Project Director should insure that the following points have been completed:

1. Power installed and tested.

2. Air conditioning installed and tested.

3. Control devices (humidity and air conditioning) installed and tested.

4. Insure delivery of necessary tape reels.

5. Follow-up shipping details with IBM Traffic Department regarding arrival date of machine.

6. Follow-up on arrival of customer engineers.

About four months prior to installation of your machine, it would be wise to publish an "EDP Status Report" for your management and supervisory personnel. About this time everyone is getting anxious, asking a lot of questions. Everyone wants to know when his application will go on the machine. You can gain a great deal by doing this. I have a copy of our status report. The first part of the book contains a summary of the EDP status which could be easily understood by all supervisory personnel. We made about 100 copies of the summary but a very limited number of copies of the whole book.

Ladies and gentlemen, it's been a real pleasure to be here today. I trust that the information we discussed this morning will be helpful to you in the event you should install an EDP machine. Your pre-installation planning chart can be used as a check off list of things which need to be done and which need constant attention. I want to repeat again, that you should determine what your objectives are as soon as you can and continually strive to attain these objectives. A computer has to be of paramount interest to top management--and management must start with a definite point of view on the final results it wants to accomplish. It's not too difficult to install a successful computer operation once worthwhile objectives have been determined, provided the people responsible for the installation are not afraid to make decisions, to sweat a little bit, and never to lose sight of the objectives.

I leave you with this thought; you may buy the most expensive machine on the market, but it will be no better than the people behind it.

THE ORGANIZATION AND OPERATION OF AN ADP INSTALLATION

Carl B. Barnes

Carl B. Barnes is Director of Personnel for the Department of Agriculture. Immediately preceding this appointment he was Director of the Operations Analysis Staff of the Agricultural Stabilization and Conservation Service in which capacity he was responsible, among other things, for automatic data processing and manpower utilization. With 25 years of Federal Service, 19 in the Department, he has had extensive experience in management activities, notably classification and placement.

The subject today, according to my instructions from the USDA Graduate School, is the Organization and Operation of an ADP Installation. From the outline suggested to me under this subject, a more appropriate title would be, "Organizational and Personnel Aspects of an ADP Installation." Today, I am assuming that you, as a Federal Government executive, are representing an agency which plans to rent or purchase a computer or which already has one. In this capacity, I hope that you will think along with me while I attempt to discuss the following eight things:

1. The main functions of an ADP installation.

2. How best to organize those functions first, within the ADP installation, and second, the relationships of these functions to other functions in a typical government agency.

3. The main configurations of equipment necessary to carry out the functions.

4. The necessary staffing pattern to plan for, manage, and operate the equipment.

5. The appropriate duties and Federal pay classifications of the key positions involved in our staffing pattern.

6. The types of recruitment problems and related selection and placement activities which might be encountered in filling our key positions.

7. The types of training programs available in order to get the new staff into quick efficient production.

These seven things I would prefer to discuss by more or less thinking out loud about them with you. This, I hope, will create enough of an informal approach so that you may feel inclined to take over the eighth and perhaps most important point. That is to raise questions about what I have said and to discuss alternative approaches of which, may I hasten to mention, there are many.

Remember, there is no magic formula for the most effective ADP organizational structure. There is no formula for what constitutes the best configuration of ADP equipment. The configurations and capacities of equipment are changing rapidly. Fantastic technological improvements characterize the computer industry. The announcements of new equipment developments this year or even this month may make many things I am saying on this point obsolete very shortly. The classification of computer positions and the staffing of our hypothetical computer operation is simpler. But, even here we find official CSC standards recognize that because of the "rapidly changing nature of the occupational area, frequent re-study and modification of standards may be expected to be necessary." Incidentally, I urge you to read these Civil Service Commission standards. Aside from representing an excellent product technically, they include much general information on the subject of ADP. And, they are written in easily understandable English. I am using the substance of those standards widely in this talk today.

Let's get to work on our eight point program.

I. The Main Functions of an ADP Division

Stated very briefly, the main functions of an ADP installation are first, to conduct feasibility studies; second, to formulate and define problems; third, to design systems; fourth, to develop programs for processing those systems through an electronic computer; fifth, to supervise the production of output through computer operations; and sixth, to optimize and standardize what we have done in the first five steps.

A. Let's talk first about feasibility studies. This simply means the systematic review of processing requirements. It usually precedes and provides a basis for a decision on whether or not a computer is to be installed or whether an operation is to be put on the computer. It involves analyses of the volume and types of data to be processed. Feasibility studies explore the cost of present operations, alternative possibilities, cost benefits which could be realized through installation of computer processing. They evaluate work flow modifications which might be involved and the abilities and cost of various models of computers for doing the job. The feasibility study in the final analysis provides the basis for determining

whether or not to install a computer. Every step subsequent to this is a further refinement of the feasibility study.

B. Now, let's look for a moment at the formulation and definition of the problem. This involves detailed consideration of the end-product data desired or required, the sources of such data, and the means of channeling data into the machine unit in forms suitable for processing.

C. The design of the system is next. This involves the development of an ideal processing plan to carry out the plan efficiently. Included in systems analysis is the detailed study of every related job, paper and document, and processing operation in the current organization; consideration of their validity in relation to data requirements; consideration of opportunities for generating other needed information or eliminating present steps, processes, and information. The systems design stage as well as other stages in this entire process provide a certain degree of check and balance on previous steps which have been taken. For example, in the systems design stage the systems designer may still decide that other than automatic data processing methods may prove to be more effective. The result of the systems design process is a group of procedural and general flow charts and functional block diagrams which express a machine operation and products in generalized terms but do not go into detailed machine steps.

D. The next step is programming. It takes the generalized systems flow charts and refines them into detailed flow charts, then into completely detailed machine logic charts. Logic charts prescribe every machine step necessary to carry out the entire operation and finally translate such steps into machine language code. Programming also involves allocating memory spaces in the computer, assigning on-line and off-line equipment by logical designation to utilize the equipment most efficiently, and determining the sequence of steps in reading, processing, and writing out. It involves planning in complete detail the integrated operations of all system units through to the establishment of machine instructions to control the file printed format. Programming offers substantial opportunity for judgment in all of these steps, because this is the step designed to obtain maximum machine efficiency and also to perform the greatest number of processing steps with the fewest machine runs. Here, as in the other steps, is room for recommending systems modification, to improve machine utilization or the value of output data. This is a step which requires completely detailed thinking in advance for the machine. Plans must be

made for all combinations of data and situations which may arise and methods are worked out for recovering the program (this means saving as much as possible of the data already processed and finding a valid re-start point) from various possible stop or error situations. Testing programs previously developed, then debugging them, also generally fall under the programming phase of our operations.

E. The fifth step is the production operation which includes the final preparation of input data, the actual operation of the computer equipment in carrying out the program and the necessary preparation of out-put data in final form by means of punched cards, punched paper tape, on-line communications systems, high speed printers, and/or tape to card converters.

F. The last step--at least within the context I am following to explain it today--is optimizing and standardizing. This includes the optimization of written programs and the development of standards to improve what has been done. This normally takes place after the production is under way. Included in this step are such things as reviewing and evaluating existing programs for possibility of improvement such as by developing standard utility programs and sub-routines and the application of improving program techniques; the development of improved or standard methods and practices of such procedures of block diagramming, programming, desk checking, test data preparation, debugging, etc.

II. ADP Organization

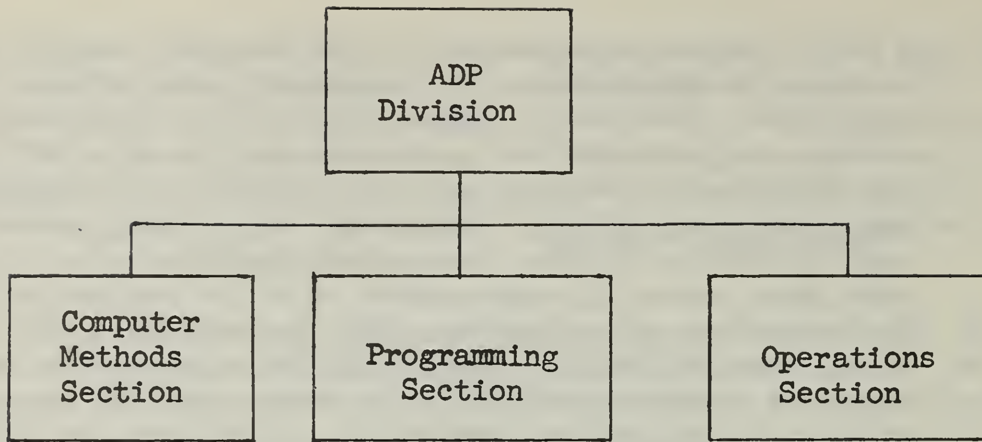
A. Internal Relationships. How do we organize these functions? First, let's visualize these functions as having somehow to fit within the framework of an organization chart within our agency. Second, let's look at the organizational phase of an ADP installation from two viewpoints. The first is the internal organization of the ADP function. The second is the external relationship of the ADP function, in terms of its basic relationship to the rest of the organization.

In discussing the organization of the ADP function, I will assume that you are renting the computer and related equipment rather than purchasing it. The main difference, incidentally, in the organizational structure of a purchased or a rented computer is that, in the former, you may want to include on your own ADP staff, engineering technicians who are capable of maintaining the equipment in effective working order. Rented equipment usually includes maintenance services by the renting company.

A well defined organization in any business, and in the Federal service, in particular, is a necessary prerequisite for efficient operation. Obviously, responsibility for all functions as well as for all positions and personnel must be well defined within the ADP group as well as in its relation to the organization of the agency as a whole. Obviously, also, there is an imperative need in ADP as in other management functions, that its management have authority commensurate with its responsibilities. But it is perhaps not so obvious that in organizing ADP responsibilities, the importance of building the organization around the types of personnel available to operate the ADP installation is of almost transcending importance. I believe the principle of building an organization around the capabilities of people to run the organization is a major factor in any successful organization. I say this in spite of the many arguments I hear which favor building organizations around functions and then obtaining people to operate these functions. I just don't believe that in the government or in industry it usually works this way.

Most of the people who have been around for some time in any organization--private industry or government--are people with known abilities, known strengths, known weaknesses. Management generally feels a loyalty to these people. These employees, in turn, feel loyal to management. Therefore, management typically tries to bring the personal as well as technical characteristics and aptitudes of its key employees into proper focus so as to get the most out of these capabilities. Today in modern management, we try to develop these capabilities properly and to build an effective team to accomplish long range as well as short range goals. We do not change all or most or even many personnel every time a major organizational change takes place. We skillfully utilize what we have, for the most part--blending in with that talent only such outside talent as, in our view, will enhance our staff teamwork effort. Therefore, I suggest that the initial ADP organizational structure in your agency be built substantially around the men and women oriented in your agency managerial needs and philosophies, whom you feel, by having observed their creative and productive capabilities, have the most potential to make it work on a sustained basis.

Now to cases. The basic organizational structure for the ADP division I will discuss today looks something like this:



It will be directed by a Division Chief. The head of each of the three sections reports to the Division Chief. The first section is called a Computer Methods or Systems Section. The second section is the Programming Section. The third section is an Operations Section. In these three sections, we have accounted for the six primary functions of the ADP installation. Feasibility studies, problem formulation and definition, and systems design will be placed in the first section. Programmin goes to the next section, and production goes to the Operation section. Optimizing and standardizing is placed in the Programming section. Let's look at each of these sections in more detail.

1. <u>Computer Methods Section</u>. The first section, a <u>Computer Methods or Systems Section</u>, would involve first the responsibility for applications analysis. This is simply the conduct of studies of agency systems and operations for the purpose of determining the feasibility of processing data involved in such systems and operations through the use of automatic data processing equipment.

   The second function in this section--Systems Design--involves the study and analysis of office methods and procedure for which the application of the ADP system has been approved. This is different from the applications or feasibility type analysis just mentioned. This involves the development of methods for processing these applications in the ADP system. This section would have to develop and define the ADP methods to be used, the overall system logic and flow diagrams. It would, perhaps, establish the number of ADP runs required, define and develop logic diagrams for the individual runs, establish specifications as to the type and format of in-put and out-put media, and develop methods and specifications for

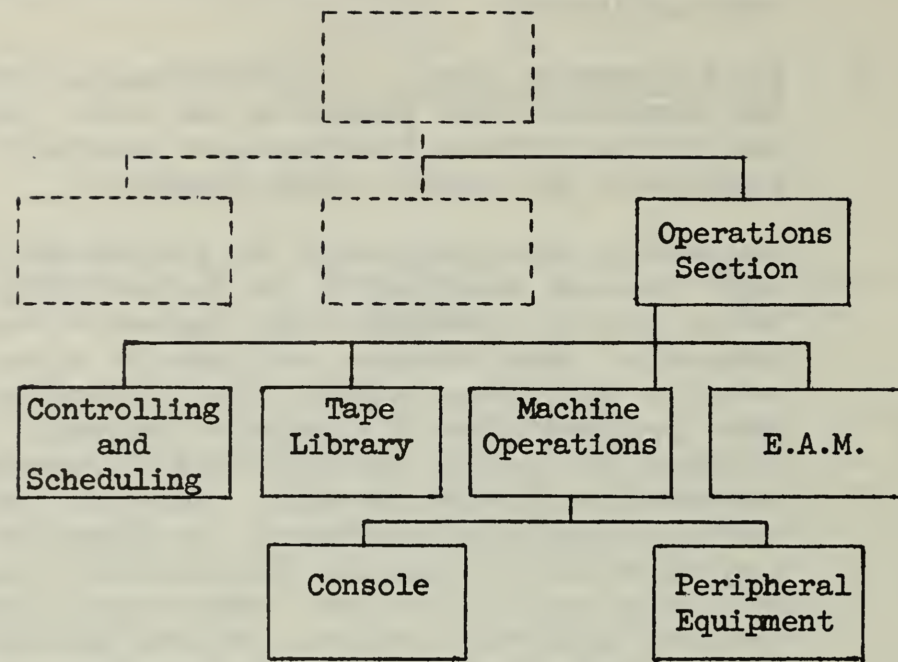
auxiliary punch card equipment operations required for each application.

2. The Programming Section. The Programming Section includes the responsibilities implied by its title. But, let's discuss the two primary functions more precisely. They are programming and program standardization.

   Programming means completing the problem definition and logic diagrams developed by the Systems group and translating them into machine block diagrams. From the problem definition, logic diagram, and detailed block diagrams prepared in the Systems Section, the Programming Section prepares programs coded in a symbolic language. This is the language the machine can understand. Programmers perform pre-assembled program checking. They test out all programs and issue completed programs to the Operations Section. They establish and maintain schedules and progress charts for all programs in process. They prepare complete machine operator instructions for the programs written including a list of "halts" and the action to be taken by the operator at each "halt."

   The second main function of the Programming Section is Program Standardization. This involves the review and evaluation of generalized utility programs and sub-routines available from equipment manufacturers and others. Programmers review and evaluate applications that are developed by systems people to determine the need for and the nature of standard utility programs and sub-routines. As necessary, the programmers develop new standard utility programs or sub-routines for use by systems analysts and programmers. They also develop standard methods and practices for such procedures as block diagramming, programming, desk checking, test data preparation, debugging, and preparation of operator instructions.

3. The Operations Section. The Operations Section has the primary responsibility for controlling data flowing into and out of the processing equipment; and for effectively operating the particular configuration of equipment necessary to process that data. I have grouped these functions into four subordinate units as follows:



The Control and Scheduling Section maintains and executes daily operating schedules and develops new and improved scheduling and control procedures. This section schedules data through the Tape Library to the computer. This involves preparing for each application, daily, periodic, and long range machine schedules of work to be processed in the ADP Division. This section assigns to the schedule, jobs to be processed in the central processing unit and auxiliary equipment and allocates estimated time for completion. It makes necessary adjustments to schedules to assure maximum efficiency for machine utilization. It prepares and issues instructions for disposition or distribution of out-put material.

The Tape Library maintains a library of magnetic tape reels or decks of program cards, flow charts, listings, program specifications, operating instructions, halt listings, runbooks, wired control panels, and related records. It maintains records of each reel of magnetic tape and a history of its use. It issues to and receives from operators tapes and card decks. It maintains, issues, and accounts for supplies and materials such as tapes, blank control panels, printer carriage control tapes, etc. In other words, it provides to the systems people and to the programmers

the material necessary in conducting many of their operations.

The Machine Operations Supervisor directs the operation and maintenance of data processing equipment. This unit conducts volume tests, using actual production data, of ADP programs which the programming section submitted. Employees in this section test routines and sub-routines of programs and machine operator instructions for validity. They supervise the operation of the central processing unit and connecting auxiliary equipment. They interpret the console lights, recognize and evaluate reasons for machine stoppages, and take appropriate corrective action immediately where possible. They see that the various auxiliary equipment connected to or operated independently of the processing unit is effectively operated.

4. Some Alternative ADP Organizational Possibilities. The organization of the ADP function within each of our agencies will vary considerably depending upon where we are, what our philosophy of organization is, and what people we have to staff our installations with.

One of the primary arguments today in considering the proper organizational structure for an ADP function is whether the ADP Systems function should go under the head of the Data Processing Division or whether it should be operated on a par with and independently of the ADP operation itself. I have included the function separately in our ADP Division today because I believe it strengthens the role of the ADP man in the organization. It gives him complete control of the things he needs with which to do an effective job and puts him in a position where he is ultimately and completely accountable to management for doing an effective ADP job.

Another argument is that the control and scheduling function should be established as an independent section within the ADP Division. This is an important function. I have included it, however, in the Operations Section because I want to have someone completely accountable to management for effective operations including scheduling and controlling.

Still another argument is that the function of programming standardization might well be combined with the function of research into new types of equipment, new systems approaches, etc., and established as an independent section within the Data Processing Division. Still another viewpoint is that

the ADP Systems and Programming Sections be combined into one unit. Actually, as the size of the ADP operation and the equipment configuration it contains tends to get smaller, this combination of functions can be found more frequently than in the larger installations. Personally, I believe that even in the larger installations, there will be a greater tendency in the future to combine systems with programming. The reason is new developments in the field of programming which will reduce to automation many of the present duties performed manually by programmers. Programmers will, therefore, gradually get more into pure ADP systems development and less in the present day concept of programming.

Other schools of thought exist naturally in this fast growing field as to what form of organization is best. There is no general answer to every situation. Each organizational structure must be tailored to fit a given situation.

B. External Relationships. Now, let's get to what, in a sense, may be a much more important aspect of our ADP organization, at least from the viewpoint of the Federal executive. That is the external relationships of the ADP function throughout the entire agency.

1. The Departmental Level. At the departmental level, most executive departments of government which have gotten in any degree into the ADP function at this point, have accounted for this function at the central staff level either in a data processing coordinating office or they have combined this function of data processing coordination with related management functions. The latter usually involves a combination of ADP with systems analysis and other paper work management functions. Since there are very few computers located in the central or secretarial office of an executive department, this function consists largely of staff responsibility to the top management of the executive department and of providing a degree of coordination and leadership on the data processing function to subordinate levels of the agency. The purpose here is to assure some degree of adherence to basic standards of effective acquisition and utilization of the expensive equipment involved in the ADP process.

At the departmental level, the importance of a watchful eye and a helping hand on the ADP function will become increasingly important to the top management of the agency. Vast

sums of money are involved in an ADP process. Centralization of record keeping is perhaps one of the primary bases for economical utilization of data processing equipment. Material possibilities exist for the utilization of such equipment across agency lines. The effective implementation of such a concept must involve careful coordinative efforts from some central level. The problem here is for the secretarial level in executive departments to realize this type of department-wide potential in ADP and then to staff his central ADP office with people of the degree of technical qualifications and leadership capacities to make the department-wide effort effective.

2. <u>The Bureau Level</u>. At the bureau or agency level within the executive department, a variety of organizational structures exist but one pattern is becoming fairly common. That is that the ADP function, in those bureaus which have computers or which intend to get them, is established on a par with other primary management functions of the agency. For example, if there is a deputy or assistant bureau chief in charge of management, the ADP function would rank along with the fiscal function, personnel function, budget function, and other management functions which are given a high degree of importance in achieving overall management effectiveness.

   In agencies where the data processing function is largely statistical or scientific in nature as opposed to the pure business management data processing type of activity, the ADP function is sometimes merged with regular operating responsibilities and it does not fall under the direct supervision of the management arm of the agency head. In all cases, however, there must be the closest type of relationship between the management activities and the scientific activities if both process data through the ADP installation.

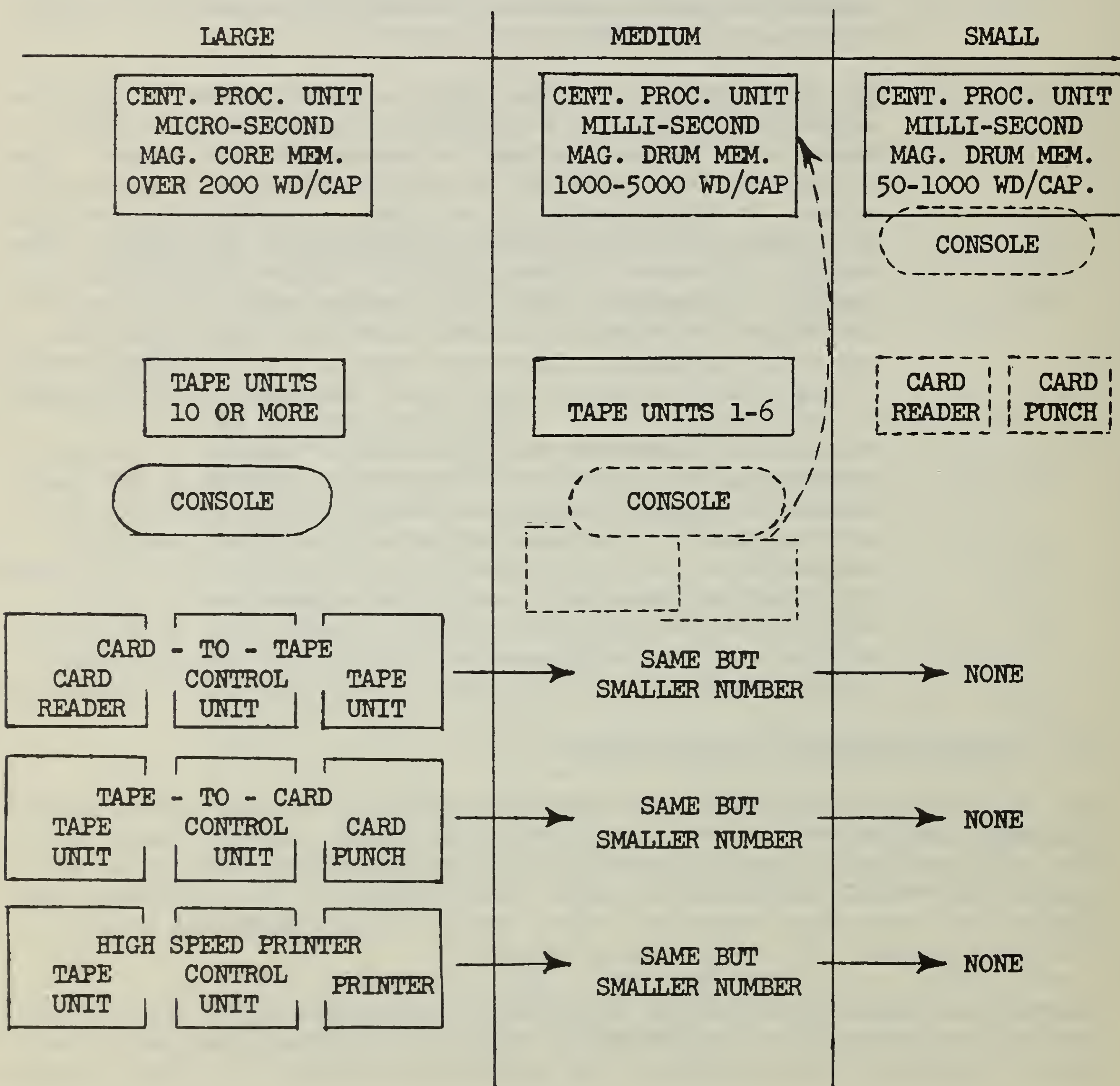
III. <u>Basic Equipment Configurations</u>

The third main heading under our eight point program is to describe briefly equipment configurations. We have already discussed the main functions of the data processing office and methods of organizing these functions into a typical government organization chart. Now, we have to fit equipment configurations into these functions, then, staff the equipment and related functions through the Civil Service process.

There is no pigeon hole process of defining configurations of ADP equipment. Any definition involves a considerable degree of arbitrary judgment. As a basis of general understanding, I am generally following U.S. Civil Service

Commission position classification standards on this subject. These standards describe three basic size groups of equipment. There is a large installation; a medium sized installation; and a small installation. As the size of our data processing installation varies, our organizational structure and staffing pattern will also tend to vary.

Here is an exhibit which gives a concept of three different sizes of equipment configurations.



A. The Central Processing Unit. Each of the configurations involves a central processing unit. The difference is that the large CPU has a speed of microseconds while the medium and small perhaps have millisecond capacities. The large has magnetic core memory while the medium and small perhaps have magnetic drum memories. The large typically has over 2,000 word capacity while the medium has 1,000 to 5,000 word capacity and the small 50 to 1,000 word capacity.

B. Tape Units. The large configuration has 10 or more tape units, (usually more), the medium 1 to 6 and the small usually would not have any tape units. It would be fed by punched cards.

C. The Console. All the configurations have a console. The large configuration represents the console separate physically from the central processing unit. The medium represents either a separate console or one built into the CPU. The small almost always involves the console as a physical part of the central processing unit.

D. Off-Line Equipment. The large size computer installation will utilize one or more card-to-tape off-line configurations, each such configuration involving a machine that reads cards and records data on magnetic tapes. The tape-to-card configuration involves a machine that reads tapes and punches data into cards. And the high speed printer configuration produces printed reports for management's use from data recorded on magnetic tapes. In the medium size installation, the card reader and punch are frequently used as on-line equipment. And, in the small configuration, such is predominantly the case.

IV. ADP Position Classification

Let's begin the process of establishing our position classification structure. This is necessary prior to staffing the positions with qualified people. Your personnel offices are naturally available to perform the task of defining what duties you want to assign positions within this structure. But, perhaps a brief review of the nature of the duties of the various types of positions involved will be helpful to you in working with your personnel people and understanding some of the main requirements.

A. The ADP Occupational Structure. The ADP occupational structure comprises five separate series of positions. They are summarized as follows:

1. The Digital Computer Systems Administration Series involves responsibility for supervising overall digital computer

tabulating equipment or performing staff planning or administrative work which requires technical knowledge of computer operations and management.

2. The Digital Computer Systems Analyst Series involves responsibility for supervising the design of or of designing systems for the electronic processing of subject matter data.

3. The Digital Computer Programmer Series involves responsibility for work in converting generalized or detailed plans or flow charts of operational sequences into flow charts for machine solution and the testing and debugging of such charts.

4. The Digital Computer Systems Operator Series involves responsibility for operating the control console of the digital computer system.

5. The Digital Computer Equipment Operator Series positions involves responsibility for the operation of peripheral equipment such as card-to-tape converter, tape-to-card converter, tape reel units, high speed printers, tape data selectors, etc.

B. Relationship of Occupational Series to Organization. How do these five occupational series fit into the organizational chart for the ADP Division?

1. Positions of Division Chief and Assistant Division Chief would be filled by Digital Computer Systems Administrators.

2. Positions in the Computer Methods Section would be filled from the Digital Computer Systems Analyst Series.

3. Positions in the Programming Section would be filled from the Digital Computer Programmer Series.

4. The top position in the Operations Section and the Machine Room Supervisor and the Console Operators would be filled from the Digital Computer Systems Operator Series.

5. Positions in the Peripheral Equipment Unit would be filled from the Digital Computer Equipment Operator Series.

6. Positions in the Control and Scheduling Section and in the Tape Library would not be filled from the Digital Computer

Occupational Series. They would typically fall under the General Clerical and Administrative Series.

C. Classification of Systems Analysts and Programmers. For position classification purposes, two series of positions are very similar not only in terms of the basic character of the work involved but particularly in terms of the grade structure and classification factors used in evaluation of the positions. These are Programmers and Systems Analysts. The CSC standards for programmer positions establish three clear cut levels of individual programming difficulty. These three levels cite types of programs (in terms of complexity) which are classifiable into Federal grade levels GS-9, GS-11, and GS-12. Beneath these levels, the GS-5 and GS-7 levels are classified as follows.

The GS-5 level programmer is classified as a basic training level. Incumbents of these positions usually receive formal classroom and on-the-job training in machine components, machine logic, command, and coding instruction, block diagramming, and flow charting, memory allocation, input and output instruction, and so forth.

The grade 7 level programmer would perhaps independently perform simple programming assignments such as coding program steps for which detailed machine logic has been developed by others. This might also involve writing machine instructions to adjust established routines to accommodate limited subject matter or specification changes, or writing insert codes for an assembly or compiler routines. In actual practice, you will find that the grade 7 programmer is usually a person who has passed his training period and whose ability is being tested by actual production assignments to assume increasingly difficult assignments.

Now, assuming that our hypothetical ADP installation is at least of medium size, I believe we can likewise assume we would perhaps have some programming work at as high a grade as GS-12. This work would probably require more than one person to operate independently at this level. From a staffing standpoint, several GS-12 people would require a supervisory position which would be responsible for planning and coordinating their work. From a position classification standpoint then, the Chief of our Programming Section would be GS-13.

We can apply, by analogy, the same line of thinking to the digital computer systems analysts. As a matter of fact, the CSC standards require the classification of these positions by

comparison to the standards contained in the programmer series of positions. So, here again, you have the grade 5 systems analyst as a trainee. The grade 7 systems analyst is assigned type of projects under close supervision selected largely to see how the individual is able to carry out progressively greater responsibilities in a production situation. Then, the grade 9 level carries out relatively routine systems assignments fairly independently. The grade 11 carries out <u>fairly complex</u> assignments. The grade 12 is either a supervisor or is carrying out what we might term <u>very complex</u> ADP system assignments. Here again, we can assume that there will be enough work of the levels of difficulty described to require a supervisor for planning and organizing purposes. Therefore, we can classify the grade of the Chief of our Systems Section in 13.

D. <u>Classification of Console Operators</u>. Now, let's concern ourselves with the Operations Section in which the various machines are actually located. Let's first classify the positions of the supervisor of the main equipment or the console operator.

The actual operation of a digital computer system console during production runs is a very important responsibility. The requirements of a successful console operator are very unique, very demanding, and very difficult to fill satisfactorily. It is a peculiar position with peculiar requirements with relatively little precedent for purposes of comparison in the Federal service. Remember that the entire production of output from our computer system is controlled by means of a control console. These consoles vary in details of appearance, control switches, number and coding of neon lights, exact method of operations, etc. In addition, differences result from the make and model of the computer, and the amount and type of equipment which is being used "on-line" in the data processing operation.

The responsibility of console operators in stop or error situations has many variables. The Commission distinguishes two levels of operator responsibility in this connection. First is <u>normal</u> operator responsibility and the second is <u>senior</u> operator responsibility. In either situation the operator is expected to have sufficient knowledge of programming to read, understand, and apply the program he is operating. But the senior operator is expected to understand program <u>content</u> and <u>intent</u> well enough to make certain program adjustments when necessary to maintain effective equipment operation.

To start our classification process for the digital computer systems operators, the grade 5 level is the trainee. He learns

how to operate the console and on-line components of medium and large scale digital computer systems. Any operation of the console by the GS-5 for medium or large scale systems is as a trainee and is done only for the purpose of receiving instructions. The GS-5 is under very close supervision. However, the grade 5 level is recognized by the Commission as appropriate for positions with normal operator responsibility where a small scale computer is involved. These would be typically card driven models with little or no peripheral equipment.

The grade 7 level is the digital computer systems operator, who operates the console of the medium or large scale digital computer system on production runs with responsibility for the quality of the result and with normal operator responsibility for recognizing, diagnosing, and independently acting on machine stoppages, error situations, etc. However, most of the runs produced by the GS-7 operator would be runs where most of the bugs or problems have previously been ironed out.

At the grade 9 level the console operator has independent responsibility, usually as senior operator on the shift, for the operation of the console and the on-line peripheral equipment of the large scale digital computer system.

E. Classification of Peripheral Equipment Operators. Now, we've classified the position of console operator. We still have the peripheral equipment to think about.

This equipment includes such machines as card-to-tape converters, tape-to-card converters, tape reel units, high speed printers, tape data selectors, etc. Very briefly stated, card-to-tape converters and tape-to-card converters are for converting data from punched cards to tape (either perforated paper tape or magnetic tape depending on the system) and vice versa. Tape reel units are used to hold and move tape required for any input or output operation involving reels of tape, and high speed printers are used for printing from magnetic tape directly from the computer and output reports for use for management.

There are three levels of peripheral equipment operators which will illustrate the grade structure involved in this series of positions. The GS-3 level is characterized by trainee assignments and under close supervision might typically involve responsibility for operating a substantial variety of peripheral equipment. Trainee operation at this level may include any and all types of equipment mentioned previously as distinctively associated with the computer operations and may also include

operation of equipment such as card punches, verifiers, machines with typewriter keyboard for writing on tape, etc. But the purpose of these assignments is primarily to train the incumbent to do such things as set switches, wire plugboards, read neon panel lights, mount and dismount tapes, load and recognize faulty punch cards, operate the machines, recognize and correct error conditions, and keep necessary records. In cases of the grade 3 level where positions involve a certain degree of production responsibility they are restricted to the operation of one or two types of the less complex equipment such as tape reel units, tape-to-card converters, card-to-tape converters without special circuitry, and so forth.

At the grade 4 level is involved responsibility for the operation on a production basis of at least one of the more complex pieces of computer equipment such as high speed printers, tape data selectors, card-to-tape converters with specialized circuitry, and so forth. Another example of the GS-4 level operator would involve responsibility for operating a substantial variety of the less complex peripheral computer equipment mentioned in the GS-3 level previously. In either case, the grade 4 level involves application of a good overall knowledge of the data processing computer system in relationship to the various machines in the system.

The grade 5 level is characterized by exceptionally responsible assignments involved in the independent operation of peripheral computer equipment, but these assignments would typically involve a thorough overall knowledge of a fairly large peripheral equipment operation and the relation of such operation to the computer system. It would also involve the application of the full range of peripheral equipment operation skills applicable to the system and the assumption of special responsibilities as an assistant to the supervisor or as a senior or special operator in the unit. This is the grade that you would typically assign to the expert peripheral equipment operator who has learned the machines and the relationship between the machines. His knowledge would be used typically in answering questions of less experienced peripheral equipment operators. He is generally considered to be a specialist in his field.

Supervisory positions over peripheral equipment operators, of course, are classified on the basis of general CSC standards for supervisors and would range at varying grade levels above the highest grade operator depending upon the number of operators in the shop, the nature of supervisory responsibility as assigned to the position, etc. There is no particular need to go into

the classification of these positions since the standards followed are well known to most management people in the Federal government.

Classification of Section and Division Chief Positions. Now, what does this add up to in terms of the key grade structure for our Data Processing Division? We have a grade 13 Chief of our Systems Section and we have a grade 13 Chief of our Programming Section. In our Operations Section, let's assume we have two shifts with a grade 9 console operator in charge of each shift. So, we have a grade 11 or 12 position as Chief of that section depending upon the nature of his responsibilities for such activities as planning, working with people outside the data processing shop, etc. In some installations this position is as high as GS-13. This adds up typically to a grade 14 position as Chief of the Data Processing Division involving a large scale computer operation and in some cases involving a medium size computer operation. In a few cases, grade 15 positions are beginning to crop up. But in some cases the Data Processing Chief position has not yet reached the grade 14 level. In the latter case, typically, you find an incumbent who has not yet quite assumed the full scope of responsibility inherent in the top position. I would urge you Federal executives who are planning to set up a data processing shop not to think you can economize by taking a conservative approach to classifying ADP positions. Such an approach is sheer nonsense to say the least. Start your shop with the highest grade positions you can establish. But follow this action by staffing the positions with the highest caliber people you can get.

Let's see now how we go about finding such people.

7. Staffing the ADP Division

In order to understand the problem facing prospective executives attempting to staff an ADP operation, let's examine for a moment the needs of the government generally in this relatively new occupational area.

In 1958 there were approximately 4,000 ADP positions in existence. The estimate for 1959 increased this figure to approximately 8,800, or more than double the figure for 1958. The estimate for 1960 is for approximately 11,500 ADP positions, while by 1961 approximately 12,500 ADP specialists will be needed. This is a tremendous increase within a relatively new occupational area in so short a time span.

In further evaluating the significance of this staffing need let's convert the estimates into salary costs. The 1959 estimate of 8,800 ADP specialists

cost the government approximately $30 million in salaries. The estimate for 1960 added an additional $40 million. For fiscal 1961 the estimated need will run approximately $76 million. I think you will agree that we're dealing with a very sizeable need both in terms of numbers of people and salary costs. There is small wonder that difficulties confront us when we try to find people to fill this need.

Naturally, under the Civil Service system, the first place we normally look in recruiting people is the Civil Service Commission. It is the responsibility of the Commission to conduct Civil Service examinations and to develop registers of qualified people available for positions throughout the government. The Commission makes lists of the names of qualified people available to hiring offices.

In the case of ADP positions, however, there is no specific examination covering other than peripheral equipment operators and ADP people at the GS-5 and 7 levels. Even at the GS-5 and 7 levels, we are not getting qualified ADP personnel because of an option under the Federal Service Entrance Examination. This is because ADP positions in these levels are filled from the examination largely designed for recruiting college graduate caliber people. What happens is that, from this register, the Civil Service Commission is selecting names of those who have scored particularly well in certain phases of the total FSEE examination which relate directly to characteristics required of ADP specialists. Such characteristics would largely be in fields of arithmetic reasoning, etc. Hence, from this register, we may find people who have the potential to do well in ADP but still need to be thoroughly trained in characteristics of computers, systems design, programming, etc. In addition, such people have to be trained in agency programs. The latter is often far more time consuming than the former. So, for practical purposes, we can eliminate the Civil Service Commission from our efforts at recruiting qualified ADP systems and ADP programming personnel.

For peripheral equipment operators, the Census Bureau maintains the register for the CSC and it may be possible from time to time to acquire such people from this register.

The next place we normally look in order to fill our vacant positions would perhaps be inside our agency. And here is where we find the best results. I suggest that a number of people in existing conventional punch card machine tabulating shops might well fit into the normal ADP office. This is particularly true of tabulating shops located in the same agency in which the newly established ADP operation is located. Remember that the type of data processed through the machine tabulating office generally is the same type which would be processed through the computer. Therefore, many characteristics of these data will remain the same. Differences will result from redesigned systems and methods together with

different capacities and characteristics of electronic data processing equipment.

Another area for recruitment inside the agency is accounting systems people. Such people generally do very well in systems work, particularly after they have been provided a certain amount of computer training. Another source inside the average agency is the various management divisions which make a practice of regularly hiring management trainees. These are young men and women who have been agency oriented to a certain extent and whose technical ability and personal characteristics have been demonstrated in a work situation. Consequently, their potential can be evaluated rather precisely by supervisors who have observed their work.

Another possible source for finding ADP people, in my opinion, is outside our agency and indeed outside the U. S. Government. In both cases it is advisable to determine where computer installations exist which have the same basic type equipment as does the hiring agency. I believe that it is entirely ethical to discuss with the proper officials of such agencies the needs of agencies in which we work and to try to obtain their cooperation in securing one or two of their trained people. Of course, in such cases, you must be in a position to provide the prospective employee with better promotional opportunities. These folks, of course, will be reluctantly given up by any agency. But our merit system is such that opportunities should be given these people to move across agency lines for better opportunities. In this connection by securing the cooperation of other computer installations in your recruitment efforts there are usually some people in such installations who have reached the ceiling under the position classification plan within a given agency, or second, may be dissatisfied with the geographical location or other characteristics of the office in which their present employment is located. Such people might not only desire a move but if it can be worked out both the current employer and the new employer benefits. Very often effective placements from their viewpoints as well as yours might be made.

Another possibility of recruiting, particularly young people outside the government, is to acquaint yourselves with the relative standings of certain types of people in the graduating classes of colleges which you think produce the type of people who would be compatible over a long-range period with the nature of the programs in your agency. This would involve a rather intensive recruitment program. But such a program might be necessary in the currently short computer labor market. This recruitment program would be more in the nature of a visit to colleges to discuss with appropriate officials the relative standings of young people in mathematics, statistics, business administration, etc., and to interview a number of these young people to determine their interest in ADP employment in your particular government agency. If interest does exist on the part of

such people, then, of course, they should be requested to take the Federal Service Entrance Examination.

In connection with recruitment it may be interesting to know about a few government statistics available on this subject. These are included in Harry Fite's report which was prepared for the Budget Bureau. They add up to something as follows: Of 129 agencies surveyed, 64% did their recruiting for ADP positions wholly from within, 10% recruited wholly from without, and 26% recruited both ways. After recruitment, or rather as a part of the recruitment process, 73% of the 129 agencies used written tests as definite parts of the recruitment. But of these, 61% used the tests for prognostic value while only 12% used the tests as rejection factors. One might conclude from this that those recruiting ADP people place considerable emphasis upon intangible characteristics which may not be so readily measured by written tests.

In connection with written tests, Civil Service Commission tests are available for use in selecting out of a number of candidates, those who have aptitude for ADP. These tests select people on the basis of their degree of ability, qualitative reasoning, abstract reasoning, and special perception.

VI. Training

Now, assuming for the moment that we have recruited a number of potential applicants, we are now faced with the problem of providing training for the new employees and to develop necessary knowledges and skills necessary to get them into fast, efficient production. I will not discuss orientation training because ADP employees are oriented no differently than others. On-the-job training, however, is a very important method of getting qualified employees into production quickly and effectively. This usually involves tailoring an individual training program for each employee on the basis of the person's academic and work background; his experience in your agency; his background in the field of data processing; and your specific ideas as to where you think the employee has the best potential for winding up in your agency.

One fairly common rule in training efforts is that a person who knows the computer and a person who knows the agency program should be initially paired into teams. In this manner one absorbs training and knowledge from the other. This is particularly true in programming and systems jobs.

As far as training programs offered by outside sources, there are 4 or 5 very excellent possibilities in this area. First in importance, I think, are those programs offered by machine manufacturers, particularly IBM and Remington Rand. Each of these companies has excellent training programs covering a wide variety of subjects, not only in Electronic Data Processing,

but in conventional punched card operations as well. Those who object to the emphasis I will place upon training by machine companies argue that the government should do its own training. I am of the opinion that once we have made a decision to get a particular type of machine we pay the cost of that machine which includes a considerable amount of training. Consequently, the manufacturer has prepared excellent training programs. Even though such programs emphasize primarily the characteristics of the particular manufacturer's hardware they are thorough and competently conducted. Of course, one must be objective in taking such training and should have the ability to weed out sales talk from training. My personal opinion here again is that the company is sincerely interested in training you properly. The only way their machinery will work effectively in your agency is to have well trained people operating and managing it.

Another outside training source is professional ADP conferences. I strongly urge top executives to insist that their ADP people attend <u>and participate</u> in such conferences regularly. Persons attending these conferences should be required to analyze the results of discussions, particularly from the standpoint of the applicability of points raised to problems in your agency.

I do not currently rate our colleges and universities so highly in terms of training ADP people. Even though such courses are improving, they tend to emphasize what I call the "broad brush" or conceptual approach. They are usually stimulating and educational in a broad sense but would not result in the fastest realization of production potential of individual employees.

The last and perhaps one of the best training methods, particularly for people already working in your ADP installation, is to have them regularly attend organizations of <u>users</u> of the same type equipment as yours. In this connection users of certain types of both Remington Rand and IBM equipment have organized groups which regularly hold meetings to discuss common problems. These are invaluable for practical training.

Thank you very much for listening.

PART 6

FUTURE OF INFORMATION SYSTEMS

RELATIONSHIPS WITH EQUIPMENT MANUFACTURERS

Dwight S. Ashley

Dwight S. Ashley is engaged in computer design development and systems integration for the Department of Defense. His previous experience has included various assignments with the U. S. Army Signal Corps and advisory activities to industry in the use of gas fuels. He received the BS Degree in Mechanical Engineering from the Massachusetts Institute of Technology and the BEE degree from George Washington University.

It seems appropriate at this time to note that although the majority of you are interested in the management aspects of Automatic Data Processing (ADP) equipment, there are some who are concerned with scientific applications. While many ADP manufacturers have made great efforts to develop equipment capable of application to either field, it is certainly true that the underlying problems are vastly different.

When ADP is applied in the management field we are expecting it to produce timely analysis of large volumes of data in order to permit decisions affecting methods, procedures, and policies. The value of these decisions must bear a relationship to the cost of producing them.

In the case of ADP application to scientific problems, there may be no other method and we have lost an extremely valuable yardstick. In addition many of these problems have a "one time" aspect requiring more or less "modification" to standard commercial equipment or entirely specialized design.

I wish to digress for a moment and make a few assumptions as to the type of ADP equipment we are discussing.

1. The problems involved are of sufficient magnitude that ADP equipment is appropriate.

2. No standard ADP equipment will fulfill <u>all</u> the requirements of the problem.

3. ADP equipment complex requiring in excess of 15-20% (on a cost basis) for development must be considered in the research state.

Our relationships with ADP equipment manufacturers are of necessity many and varied. They began with the initial concept of use of ADP and in many cases never end. Organizationally our relationships run through the

gamut not only of maintenance, sales, engineering, and research personel but also through that of contract administration, fiscal, legal, and management personnel. Our relationships may be personal or written, direct or indirect (as through our own legal contract administrator or procurement paths). Throughout this complex relationship we must keep in mind the two objectives sought.

1. Our objective is to obtain an ADP complex capable of doing the job at the lowest cost to the Government.

2. The objective of the equipment manufacturer is to produce a satisfactory device at the lowest cost to the corporation.

In thinking of the above statements we have ignored the obvious divergence in profit motivation. The manufacturer is entitled to a reasonable profit and in fact if he is to continue in the field must make a profit on his operations. Beware the equipment manufacturer who does not want a profit. It will show up somewhere.

While there are many ways in which we could examine our relationship with equipment manufacturers, the one based on a time scale seems to be the most suitable. With that in mind we should construct the phases of this scale.

1. Problem Analysis
2. Functional System Design
3. Procurement Specification
4. Precontractual Negotiation
5. Proposal Evaluations
6. Selection of Manufacturer
7. Contract Monitoring
8. Delivery and Test
9. Maintenance and Operation
10. Replacement and/or Modification

It would be futile to attempt to describe accurately the division of the time scale as the divisions would vary with each system; however the overall period will cover from 10 to 12 years. The useful operating period will extend for from 6 to 8 years.

The early period of "Problem Analysis" is a very crucial one in our relationship with equipment manufacturers. The character of this relationship will depend to a great extent on the position in which we find ourselves. While many variations may exist we can subdivide them as follows:

1. Replacing an existing ADP system to cope with expanding problems or to provide more economic operation.

2. Instituting an ADP system to provide capabilities not available in hand or to replace antiquated processing equipment. (This is the position of many industries contemplating automatic process operations.)

3. Developing system approach to problems involving ADP where the problem parameters and solutions are vague and uncertain.

4. Integrating ADP philosophy into a partially automated process in order to provide increased capability and/or more economic operation.

In cases 1 and 4 we could expect to find a knowledgeable group within our own organization. They would be familiar with appropriate ADP equipment and should be able to analyze their problems and develop a functional design in a minimum of time without recourse to equipment manufacturers.

In case 2 an entirely different situation exists. No such group will be available. This organization is in an extremely difficult situation and should seek a source of impartial advice such as another experienced government organization or an unbiased commercial concern.

Even with this aid they will have to state their projected problems in terms which are understandable to those in the ADP field, they will be required to estimate future expenditures for budget purposes, and will ultimately have to prepare procurement specifications.

Armed with all the advice available and maintaining an <u>open</u> mind, the organization should contact <u>all</u> reputable equipment manufacturers. Initial contact should be made with the representative (Salesman). This man is the one most qualified to speak a language understood by the organization in question and still able to translate the problem into appropriate ADP Language for <u>his</u> company.

We should not under-rate the value of the company representative to us. He brings up-to-date information as to new developments in commercial organizations and often news of future work and research direction. Much of the information given is of a company proprietary nature and it is mandatory on us to keep these confidences inviolate. Signing scraps of paper stating "that the information will not be released, etc." is of no value; the obligation is strictly a moral and ethical one and only the highest integrity in this matter will suffice.

The representative also serves as an intermediary between the engineering groups in his company and the Agency, supplying the necessary detailed information on his product to permit its incorporation into an ADP system. Contact with the representative will save much travel and time both on the part of the Government and the equipment manufacturer. We must always remember that useless costs incurred by a commercial organization are reflected in higher costs to the Government in the "overhead" rate.

In his work the representative of the equipment manufacturer is exposed to a wide variety of problems and has solutions to some. We should listen to his advice and use it when applicable. We must be continually alert to avoid being "sold." Under no circumstances must we be in the position of making a decision based on contact with a single equipment manufacturer.

At this point I wish to bring a note of caution into this phase of activity. When an agency or organization indicates interest in ADP and goes to the extent of discussing its problem with an equipment manufacturer's representative the agency become a "ripe" prospect. He will attempt to bring all types of sales pressure to bear. Many books have been written and courses taught on salesmanship. Salesmen range from the schools of "Madison Avenue" and from the retired high ranking positions in the armed forces to ex-vacuum cleaner salesmen and recent graduates.

Entertainment has become an accepted sales technique and extends from a lunch to shooting trips in Canada and golf in the West Indies. All this entertainment is covered by expense accounts and in many cases becomes embarrassing or impractical to refuse. To attempt to establish rules under such varied conditions is futile. Let me just say this--be prepared, willing, and <u>do</u> return all accepted entertainment in kind out of your own pocket. This may seem unfair, mainly because it is, but that is the climate in which we find ourselves and it is the only way in which we can maintain our impartial attitude with a clear conscience.

Perhaps we can sum up our relations with the equipment manufacturer sales representative. He provides information on equipment and developments, relays engineering information, and has some knowledge of similar problems and solutions. We must exercise extreme care with proprietary information, contact more than one representative, and avoid personal obligations.

An organization in the position of that of developing a system based on vague and unknown parameters is faced with the problem analysis area and the system development. Two avenues of approach are open:

1. Develop a group capable of defining and implementing the System Design.

2. Contract with another agency or a concern which has this capability but which does not manufacture equipment.

Both of these approaches are difficult of attainment but no other solution exists. In any case no direct contact with an equipment manufacturer is warranted or advisable.

When we finish the phase of analyzing our problem we approach the solution through the step of "functional system design." Although this

design has been developing in our minds all along, it is at this point that we must formally commit it to paper as an integrated entity. A great many preliminary decisions must be made in order to obtain a practical ADP system.

These decisions involve breaking down an idealized design into areas in which ADP equipment may be procured, those in which a certain amount of development may be necessary, study of all areas in which man-machine relationships are concerned, compatibility requirements, and finally a clear understanding of the existence of "bottle necks" with possible solutions and the resulting consequences.

In addition to the technical problems of design and programming, the area of economics becomes important. We must ask ourselves what ADP is going to do for us and regardless of the difficulties involved set a value on its use. We must determine the cost of our proposed ADP system over its useful life. We will then be in a position to provide proper budget justification.

During this period we will have been in constant touch with the engineering, programming, and applications groups of many equipment manufacturers. These contacts will for the most part be informal and on a professional level with the mutual objective of solving a problem.

I stated that these contacts are informal and feel that this is the only fair way to approach this area in that we have not as yet decided on a procurement procedure and have not prepared any definitive specifications. We are gathering information and this should be made clear to the equipment manufacturer. There is no question as to his cooperation as he fully realizes the importance of his participation not only from our standpoint but also from his own.

There is one thing we must keep in mind. We must not abuse this cooperation to the extent of winding up a contact with "let us have your proposal and we will consider it." No request for a proposal should be made until we are sure that such a proposal will be responsive to a reasonably firm specification and in accord with our individual contractual procedures.

We are now presumably in the following enviable position.

1. We know how ADP will solve our problem.

2. We know what ADP system we should procure.

3. We know the value of such a system to us.

If we have designed the functional system properly there are still

several degrees of freedom inherent in the layout. These variations must be incorporated in the next process of specification writing in such a manner as to provide some latitude for engineering development.

We must take great pains with the preparation of specifications. They must describe our intent exactly and in such a manner that no questions can arise as to fulfillment of the requirements.

I have used the plural of specification because any large ADP system may include one or more of the following:

a. Purchase of standard equipment.

b. Rental of standard equipment.

c. Development of special equipment.

d. Provision for commercial maintenance.

e. Operation and/or programming facility.

To include all of these requirements in one specification is patently unfair to all except the largest ADP equipment manufacturers. To procure in this manner would effectively stifle competition in the field and ultimately result in our complete dependence on a few large suppliers.

The specifications should be prepared for each of the appropriate areas. They should be carefully written to describe what is to be done and when it is to be completed. Great care must be taken in their preparation that nothing is included or specifically excluded which in any way could be construed as applicable to one or more equipments of specific manufacture. Enough freedom should be allowed to permit advantage to be taken of certain features of various manufacturers.

Preliminary specifications should be drawn up and together with "letters of interest" forwarded to <u>all</u> reputable equipment manufacturers. This should result in meetings with each interested manufacturer or supplier in order to clarify any vague points, modify detailed specification where necessary, and to reach <u>full agreement</u> as to requirements. Based on the sum of these meetings final definitive speceifications may be prepared and forwarded to each interested manufacturer for his formal proposal.

Although the purpose of this procedure has been to prepare a good set of definitive specifications, there is little doubt that many questions of contractual nature have been raised. Our contacts have probably been with all levels of engineering and management. Careful documentation is required. Our own contract administrators and their legal advisors should be included in appropriate discussions.

The questions in patent rights, background patents, association with other manufacturers, long term operating and programming requirements, company participation in development, etc., may necessitate policy decisions and the appropriate persons should be available.

The type of the contracts to be considered, such as fixed price, standard procurement, cost plus fixed fee, time and material, rental, etc., should be discussed.

It should be made very clear that in the case of CPFF contracts, the bids must be realistic and no overrun in dollars or delivery date will be considered. I make this statement with some trepidation as I fully realize that there are uncertainties in development (not research) work and that overruns are sometimes justified. A lack of firm policy in this area has resulted in manufacturers depending on these overruns to cover their costs and bidding low to obtain the contract. This practice is intolerable--resulting in poor budget planning, lack of respect for the manufacturers' integrity, and a general collapse of good relationships.

I do not know the solution to this problem of overruns, but can suggest the following:

1. A percentage sharing of overrun costs.

2. A ceiling on costs based on the average of bids submitted when a lower bid has been accepted.

3. Penalty for late delivery together with incentive bonus for early delivery where an advantage to the government accrues.

During this period is the time for the development of mutual respect. A thorough knowledge of our own problems and an ability to express them in terms which are understandable to the prospective contractor is a necessity. He will be unable to present a proper proposal without this information.

We must insist that the proposals to be made are responsive to the requirements and provide sufficient technical detail to permit comparison. We must remember that problems involving compatibility between different manufacturers' equipments require resolution in the proposals. At this point let us keep one thing in mind. We have asked the equipment manufacturers for a good proposal. We must give them adequate time to prepare a good one. Too often we try to hurry them at this stage.

We have now reached the point in the program where a number of formal proposals have been received and we are faced with the task of selecting the best from all standpoints.

To make an accurate evaluation of the proposals is one of the hardest jobs we have to face and there are no rules that we can follow.

However, some suggestions may be in order.

1. Reduce all proposals to the same base from a monetary standpoint. This will include such items as professional labor, production labor, appropriate overheads, materials, sub-contract, G & A overhead profit, etc. Make sure that the costs of each proposal are reflected in the same category.

2. Examine the technical portion of each proposal to insure that the requirements of the specifications have been met. Not all proposals will approach the problem in the same manner. In some cases plus features will appear. In others there will be certain deficiencies in quality tolerances, etc.

a. Although our specifications have taken into account predictions as to future requirements we would be well advised to look into the capabilities for expansion included in the formal proposals. This is an important feature in the event our projections have not been adequate or unanticipated increases leave us with an inadequate system.

b. All areas of the complex should be examined carefully from the standpoint of reliability. The inclusion of checking features programmed or other correction circuitry, marginal operation capability, and the use of proven components well derated, are some of the items which should be included where applicable. That portion of the complex where development work is necessary deserves an even more thorough scrutiny.

c. The ease with which an ADP system can be maintained is equally important whether we contract for the maintenance or provide our own labor. The costs are not only those of labor and materials or contract but the "down time" resulting. With this time valued in the hundreds of dollars an hour the saving resulting from ease of maintenance is apparent. Maintenance facility is not a matter of luck but is inherent in the design of the equipment. We should examine the entire system proposals for these features.

d. Ease of operation is of primary importance to any system. This is the point of contact of the system with the outside world. Man-machine relationships are important. A careful separation of operating and maintenance functions is desirable. A word of warning is in order at this point. The equipment manufacturer is aware that the operating features are his main selling point to the user of the equipment. He is likely to go overboard in this area of design

before the functional aspects of the system are firm and few of these original paper designs find their way into the final equipments.

Here we must carefully evaluate a design keeping in mind the fine balance between necessity, desirability, and operator whim.

It is expensive to provide fool proof operating features but impossible to make them "damn fool proof." Great care must be taken in examining this portion of the design.

e. Features which tend to simplify the programming and debugging problem should receive great attention and their inclusion in a proposal regarded favorably.

f. Examine each proposal from the point of view of installation, proper site environment, facilities, layout, etc. to insure that all requirements have been included.

g. Our original specifications included a section on acceptance tests. We must be careful to verify that a full understanding of these tests is reflected in the proposal and that <u>all</u> equipments regardless of the method of procurement (rental, purchase or other) are responsive to the procedures outlined by the acceptance tests. By this I do not mean to imply that all elements are of equal reliability but that all elements have the required reliability to accomplish their normal function.

At this point it would be very valuable to list the proposals in order of a technical evaluation of "how" the manufacturer intends to implement the system design. A crude weighting of the points we have discussed may accomplish this evaluation. Statements that certain requirements of the specification will be met without describing "how" should receive "0" weight; where we are uncertain in our judgment as to the relative merits of certain solutions, equal value should be given.

After arranging the proposals in ascending order of technical merit we should add the normalized cost proposals broken down as has been suggested.

If our evaluations have been well done and based on the old saw "you only get what you pay for" there should be no discrepancies in the ordering of the proposals. If any exist they should be carefully re-examined to ascertain the reasons. All proposals should now be re-examined starting with the one having the lowest total cost for full compliance with specifications, with particular attention to those areas indicating complete and mutual understanding of the work involved. Great care must be taken to insure that the costs shown in the proposal are realistic in view of

the technical requirements. The reasons for the lower costs must be determined. We must keep in mind that the lowest cost proposal is more likely to overrun, all other factors being equal.

If any proposals can be eliminated after this re-examination we must continue until we reach the one that cannot be eliminated. This proposal is then selected for negotiation. If there is little cost difference between the one selected and the next higher one we would be wise to re-examine that one in light of possible increased capability in some area.

However, unless something important has been overlooked in the first evaluation we should accept the lowest cost proposal that completely fulfills our requirements as outlined in the specifications.

The subject of contract monitoring is extremely controversial, starting with the definition of what functions are involved. In general it involves all those steps necessary to insure the delivering of equipment completely fulfilling the requirements at the time specified.

If we had complete faith in our ability to have foreseen all possible contingencies and allowed for them in our specifications, and if we had full and abiding faith in the selected manufacturer to have taken into account all contingencies in his proposal, then and only then, monitoring of contracts would be unnecessary. Not having reached this utopia we have contract monitoring to some degree whether we like it or not.

The selected manufacturer has presented a plan for implementing the required ADP system and it is one of the functions of the contract monitor to insure that this plan is carried out and that deviations from it will not endanger the quality, timeliness, or cost of the equipment. This function will require continuing contact with the engineering and production units of the manufacturer. While progress reports are usually a requirement they are almost always too late for any corrective action to be taken. In general these contacts must be on a personal rather than any written communication basis. Decisions must be made in time for effective corrective action to be taken. Many of these decisions will involve aspects of the manufacturer's operations requiring management decisions on his part and evaluation of contractual and legal implications on our part. Within the complex framework of departmental and interdepartmental contractual relationship on one hand and corporate hierarchy of decisions on the other, the question of making a timely decision with authority is of paramount importance. Often the decision is based on technical facts and the necessity to brief everybody concerned in both the government and in the manufacturer's organization incurs waste out of all proportion to its value.

This is the situation which exists today, and it is a problem to which we must address ourselves. We must find a way to put authority in the hands of the person most competent (not based on GS Grade) to make this decision.

In the case of a technical point the responsible engineer should make the decision. It is true that in most cases his decision will be reflected in all areas of the contractual relationship between the manufacturer and the Government.

We must of course review these decisions and in some cases countermand them but at least while these wheels are slowly grinding, progress will not be held up. Too many wrong decisions mean we have simply selected the wrong man for the job or the wrong manufacturer. There is one more thought on this subject. We have spoken of giving the most competent man the authority to make decisions. We must now reverse the usual relationship and brief <u>him</u> on all phases of a project from the management standpoint in order that he will have full knowledge with which to make an intelligent decision.

I hope that you will give this intolerable situation of indecision in Government-manufacturer relationship your serious consideration and recognize that the blame for much of our lack of progress in certain fields can be laid at its door.

During this phase of our relationships it is essential that complete documentation of all actions, conferences, conversations, etc. exists. If both the manufacturer and our own representatives are required to report on all phases, a mechanism exists to eliminate misunderstandings and to provide a record of any changes which may affect future negotiation regarding overruns, etc.

During the long association with the manufacturer's personnel required during this phase of the work, relationships may be built up both on a professional and personal basis. We must be careful to maintain ourselves free of any obligations.

During the period prior to delivery of the equipment we must insure that all the Government's responsibilities concerned with providing suitable areas, environmental facilities, etc. have been complete. Everything necessary to the expeditious installation of the ADP system, including any auxiliary equipments, should be provided. The manufacturer's engineering group will be putting out their full efforts on debugging the equipment and unnecessary delays will be costly.

Upon completion of the installation, acceptance tests provided for in the specification will be run to determine the fulfillment of the requirements. We should not attempt to construe these tests as an indication of reliability. Most of the components will be relatively new and a high degree of reliability should be expected.

Successful completion of the acceptance tests at the final location will normally terminate our relationship with the manufacturer. There will

certainly be details as to final reports and billings to be wound up but his job is essentially finished. However his interest in the equipment delivered is not over and if possible reports of its operating characteristics should be forwarded to him. He can learn from these and the cost is negligible.

In operating and maintaining ADP equipment we have three choices as to method:

a. Rental equipment including maintenance.

b. Purchase equipment and contract maintenance.

c. Purchase equipment and provide our own maintenance

In all cases we are concerned with keeping the two types of maintenance, scheduled and unscheduled, to a minimum.

In the cases of rental equipment and contract maintenance, these items will bear importantly on the cost of these services. With ADP equipment time running in the hundreds of dollars per hour, down time resulting from an inherent lack of reliability or poor maintenance operations is a large part of the total cost. Contract provisions limiting down time should be included.

Down time will usually drop as the personnel acquire more experience and as the initial failure of weak components is oversome.

Just as in the case of maintenance, the high cost per hour of an installation requires that the operations of an ADP system be made highly efficient.

Work plans and programs should be as free of operator intervention as possible. Every effort should be made to make the up time of the system as productive as we can. The use of off line auxiliary equipments should be included where appropriate in order to increase the productivity of the main system.

During the useful operating life of the ADP equipment there will be many areas where new developments will antiquate portions of the installation. We must make every effort to take advantage of these improvements through modification of the system where practical. These changes will allow us to extend the useful life of the system, and to a great extent enable us to keep pace with a probably expanding work load. A large part of the cost of any system is in depreciation and obsolescence, and any extension of life can easily be measured in dollars and cents.

It is some time during this period that our cycle is complete and we will consider a new system.

There is now another problem to consider in that we must dispose of our obsolete system. Many paths are open to us.

While the system may be obsolete based on our requirements, there are many areas in the government where such a system would be more than adequate. Every effort should be made to find these applications.

Many schools and universities can profitably employ such a system.

Two words of caution are in order. The entire system must be disposed of as a unit in order that the other organization may have a usable piece of equipment. Obsolete bits and pieces of hardware are of no use to anyone.

We must not consider trading in our system on a new one by the same equipment manufacturer. It would be reasonable, but not completely fair, to provide for disposal of the obsolete system by making such disposal a part of the requirements upon which all manufacturers are asked to bid.

The picture of used ADP system dealers on every vacant lot is not one which I like to contemplate.

This situation of replacement is rather new and I believe careful thought must be given to the impact on the field as a whole rather than the immediate financial gain which may accrue. I think most equipment manufacturers would agree with this position.

I have attempted to show some of the complex relationships which exist with ADP equipment manufacturers during the course of procurement of a typical ADP system. We have explored the differing objectives in these relationships as well as those which have a common cause. Much of what I have said has been to indicate the position which we in the government must be in, in order to provide effective guidance to our superiors.

In addition we have a wider responsibility to the field as a whole. To carry this out we must make every effort to take such actions as will:

a. Provide intelligent guidance as to the direction ADP systems development should take.

b. Promote increased competition in the ADP system field.

c. Develop a means of bringing together competing manufacturers whose technical developments are complementary.

d. Make every effort to provide an efficient "decision making" capability in the hierarchy within which we work.

HARDWARE PERSPECTIVE AND PROSPECTS

H. S. Bright

Herbert S. Bright, Director of Engineering Activities, Data Processing Group, Office Equipment Manufacturers' Institute, has, since 1954, been engaged in the application of general-purpose digital computers. Previously he engaged in radar system and fire-control (analog) computer engineering, and in pulse circuit and nuclear instrumentation development which led to special-purpose digital data reduction apparatus design.

Introduction

Earlier sessions in this series of seminars have reviewed basic principles of automatic data processing: evolution of punched-card machinery; apparatus for originating, communicating, storing, and retrieving data; operating principles of stored-program electronic digital computers; elements of programming such machines; automatic programming; a variety of applications of ADP; feasibility studies and broad planning considerations for facilities; detailed planning of installations; and the management of operating computer facilities.

The Planning Committee has asked me to provide you with up-to-date information about equipment developments, both those which are now on the drawing boards and those which should be anticipated on a longer-range basis. The objective of this discussion will be to bring to you information which will help you understand, in Dr. Holden's words, "...the extent to which expected future hardware developments should be taken into account in current planning for ADP operations."

At Mr. Houseman's suggestion, I will review briefly the status of some kinds of data-originating and data-handling auxiliary equipment which have been only touched upon by several of the previous speakers, and, as a postscript to Dr. Alexander's talks, I will examine in more detail present trends in design of computing machines proper. I will review for you the current status of standardization in the data processing industry, and will attempt to leave you with a little better feeling for the extent to which outlook should affect planning.

Document Reading

In the past, input to all kinds of computer and auxiliary apparatus, when taken from handwritten or printed source documents, has ordinarily required manual transcription between the original, human-to-human medium

and the primary machine-readable medium (usually punched paper, either cards or tape).

During the past several years extensive research and development effort has been expended on investigation of human-readable to machine-readable transliteration. At one extreme, large-scale digital computers have been applied to the generalized study of "pattern recognition," one special case of which is the problem of recognizing hand-drawn alphanumeric characters. (Some form of flying-spot scanner was used as an input device.) At the other extreme has been the development and manufacture of equipment for recognizing highly-stylized characters having shapes especially designed for ease of reading by means of special equipment. Some of this equipment (MICR) recognizes characters printed with magnetic ink; other equipment (OCR) uses optical scanning to recognize characters printed with conventional printer's ink.

The first publicly-recognized standards for MICR were developed under the auspices of the American Banking Association for use in automatic processing of bank checks. Only a few months ago, a great deal of public interest was attracted when several U. S. manufacturers announced the first commercial installations in banks of equipment for MICR. Application of this equipment in banks in all sections of the U. S. and in a few other countries is now increasing rapidly.

The first substantial commercial venture into the OCR field was made by the Farrington Electronics Company with a reader for decimal numbers imprinted from embossed cards through inked ribbon onto paper forms. A special font is used. As of late 1960, Farrington has a significant number of operative readers in customer hands and several are already in regular scheduled commercial operation.

An interesting step was taken just two months ago when the largest supplier of data processing equipment, IBM, announced the Model 1418 Optical Character Reader. This device, offered as an addition to the 1401 Data Processing System, transliterates into punched card or magnetic tape form one or two lines of numeric data printed in conventional IBM fonts.

A development project for optical character sensing standards was started under the auspices of the National Retail Merchants Association and was broadened last August when this work was transferred to a special subcommittee X3.1 of the newly formed ASA Sectional Committee X3 on Computer and Data Processing Standards. Subcommittee X3.1 was established for the specific purpose of developing an American Standard for character recognition. The second meeting of X3.1 was held at ASA's New York Headquarters on October 25 and 26; work is going forward on development of mutually acceptable numeric type fonts, standards for measuring optical properties of papers and inks, glossary development, and coordination of X3.1 work with that of other groups.

Work on a full alphanumeric font standard is now at a much earlier stage, and will probably not move rapidly until the numeric font work nears completion. X3.1 feels that, while the alphanumeric font is important and urgent, its completion would not be accelerated by diverting effort to it now, while that would substantially delay the numeric font.

It is my ultimate hope that a number of years hence, by dint of cooperation between manufacturers and users of data processing machinery, standards will have been generally placed into effect which will permit such equipment to read source documents which have not been prepared in any special way. This should greatly improve the accuracy and timeliness of many kinds of business data processing operations. "When," you ask, "will this concept extend to printed matter in general?" That, gentlemen, is much farther away. Where needed, I feel that equipment to read such material will be available, but will remain "special" for the foreseeable future.

I should like to offer here the opinion that document reading will find its real utility in applications where record-keeping has been inadequate because of the use of manual transcription between two human-to-human media, in particular in cases where a printed or typewritten document has been transcribed at least in part to a hand-written record. I do not believe that large-volume keypunch-and-verify operations from hand-written sheets will be changed, because that process is normally high in both accuracy and economy.

Data Handling

In Session 3 of this seminar series, Messrs. Malone, Griffin, and Schmitz spoke to you on the subjects of source data automation, data transmission, and the storage and retrieval of reference information. I will expand only slightly on their remarks, to comment on current and anticipated developments in these areas.

The work on document reading which I described in the previous section appears to have limited application in the context of source data automation. For a very wide variety of applications, punched holes in some sort of paper or cardboard medium offer a simple and economical means for performing the first step in a data processing operation, namely, placing the primary data in machine-readable form. Where substantial volumes of data are produced on a continuing basis by machines of some sort, it is in many cases feasible to produce a separate machine-readable medium in addition to printed tabulations. Development in this area seems to be moving toward improved standardization which promises interchangeability of records between machines of different manufacture, together with generally better reliability. For many purposes, particularly when a character-by-character machine is operated by a human being, present equipment is certainly fast enough.

This is not true in the case of certain large-scale scientific and engineering applications of SDA, where data may be produced at extremely high speeds, in non-repeatable experiments. In certain industrial applications, while data may change slowly, it is often desired to sample a large number of data-producing devices in such a way that all readings will be essentially simultaneous. This may be accomplished either by storage of the data for leisurely readout, or by rapid scanning of all stations from time to time. Storage may be performed either in an electronic buffer device or by punching into tape or cards at the various data origins; either process usually requires conversion from analog to digital form with a converter for every channel. The present trend is toward rapid scanning at the analog stage, conversion in a few high-speed channels, and output on magnetic tape for future processing on digital computers.

New developments in data transmission are proceeding most rapidly in the relatively modest speed ranges. While speeds of about 300 alphanumeric characters per second, transmitted over leased telephone wire circuits, have been available for several years, and while short-range transmission over special wire circuits at speeds of 15,000 characters per second and higher has been available with commercial equipment, customer interest has developed slowly. At present a number of manufacturers either have recently announced or are planning to announce data terminal equipment for use on dialed-up connections. Speeds are in the area of 100 characters per second.

In some applications, leased-wire lines are available on a part-time basis. In others, relatively short distances make leased lines inexpensive. In both circumstances, I have recently seen data transmission plans canceled because of the very high cost of buffer and control equipment.

As one approach to this economic problem, Philco has recently announced an inexpensive device by means of which the Universal-Buffer-Controller of a Philco 2000 computer installation can communicate, at speeds up to 300 characters per second over leased telephone circuits, at relatively modest terminal cost by time-sharing the UBC with other parts of the computer installation.

Source data automation is one aspect of the ways in which the entire cycle of data processing is becoming more and more automatic. Size and complexity of operations are increasing; reduction of the amount of human handling of data must proceed at least in step, or the productivity and predictability of the entire process may go down instead of up, in the face of equipment advances.

When large scale computers, with their very low unit cost of computation, first became an important factor in U. S. data processing several years ago, there developed a trend toward centralization of machine

facilities. Centralization brought on considerable interest in high-speed data transmission, together with decentralized operation of input-output equipment. At the present time, there is strong interest in the concept of the "satellite computer", which means the use of a small computer for data handling operations which are incidental to the operation of a larger computer, either locally or remotely.

By preprocessing of input data; by performing data handling operations which are tape-speed limited rather than computer-limited; and by writing of output reports and other output data processing, a satellite machine may offer substantial economy advances and may lessen or remove scheduling difficulties which sometimes plague centralized facilities. In many cases, the satellite operation permits the volume of data flow between machines to be reasonably small, so that communication facilities of moderate speed may prove adequate.

"Information Retrieval" is a term which until recently was used chiefly by workers in the field of documentation and library science. Early efforts to apply general-purpose computers in this field were costly; the only large-volume storage medium usable by the machines, magnetic tape, had to be searched completely and in a fixed sequence in any searching entry to the file. Relatively simple special-purpose devices, mostly operated manually, were quite successful.

High-powered special-purpose equipment is now beginning to be applied to this problem. The Magnacard system, developed by the Magnavox Corporation, involves small pieces of magnetic tape, each of which contains about 3000 bits of data. Magnacards, capable of being processed by special pneumatic apparatus at a speed of 90 "cards" per second, are being proposed as an element in large data retrieval systems. Automatically-searched roll microfilm files are already available from several manufacturers.

All of these hardware schemes are capable of being applied to the retrieval problem at reasonable efficiency and cost. The limiting factor at the present time seems to be the preparation of files of abstracts which are to be processed by the storage and retrieval scheme. This is still a manual process and one which, for even a small library such as commonly serves a single laboratory, may be forbidding in magnitude and in cost. The use of general-purpose computers for preparing abstracts has only begun, but already there are indications that this approach may be workable. Present work in this area is seriously limited by the cost of the transliteration process between the printed page and punched cards or some other computer-readable medium.

Trends in Computer Design

In this section I will refer repeatedly to different size classes of stored-program digital computers. For brevity, I will use four terms to

designate four general groupings of machines, in terms of their computing speed and capacity. The way in which the machines handle data internally will be used to differentiate between the groupings. Well-known machines, rather than new types, will be used as examples of each class:

Low-Power:	Serial-serial (i.e., serial by bit within each character, and serial by character within a word (LGP-30, etc.)
Medium-Power:	Parallel-Serial (i.e., parallel by bit within a character, serial by character within a word, (IBM-650, Burroughs 205, etc.)
High-Power:	Parallel binary (IBM-704, RR-1103a, etc.) and multiple-channel parallel-serial machines (IBM-705, etc.)
Super-Power:	Parallel machines with look-ahead (IBM 7030, RR-LARC, etc.)

The machine generation which is now closing was characterized by the widespread use of high-power vacuum tube machines. The start of what might be called this era was signalled by the first deliveries of the IBM-701 and the ERA-1103 (vacuum-tube, diode logic, electrostatic memory) machines; the end of this era came during August 1960, when IBM announced the availability for sale, at very large price reductions, of used 704 computers.

During the last several years of the 1950-1960 decade, while the high-power vacuum tube machines were going into service, development of transistor technology proceeded rapidly. The present machine era appears to be characterized by a remarkable uniformity in the technology used for recently-announced high-power machines. The Philco 2000, IBM-7090, CDC-1604, Honeywell 800, RCA-601, and Univac III are all parallel binary machines using resistor-transistor active logic, magnetic core memory, and fully buffered magnetic tape.

While these machines were being designed and brought into manufacture, a great deal of work was being carried forward in the development of other circuit element technologies.

Work on one of these technologies, cryogenic-superconductivity ("cryotron") logic and memory systems, has slackened during the last year or two. In the specific case of the cryotron, hopes for quick development of very designable components of fairly high speed (viz., in the small-fractional-microsecond operating time area) have not yet been realized. Early results on microminiaturized cryotronics look promising, but at present are far from commercial practicality.

Another promising hardware concept is negative resistance diode active logic, of which the practical embodiment at present is the "tunnel" diode invented by L. Esaki. The tunnel diode has shown switching speeds in the fractional-nanosecond region and is very attractive as to size, power dissipation, and high-temperature thermal stability. Application of this element is being held up by lack of design methods for circuits using TD's. The TD is a two terminal device; hence, the same terminal is both input and output, and all circuits must be bidirectional. At present, unfortunately tight tolerances are required in TD current characteristics by some of the more successful types of circuits for computer elements. Power supply (although currents are small, very large numbers of TD's are used in some arrays) and interconnection problems are quite difficult at present. One laboratory I visited recently was just completing construction of a 1 volt, 1000 ampere d-c power supply for TD circuit testing!

Although no computer manufacturer has, as of late 1960, announced any hardware using TD's, two leading manufacturers of electronic instruments (Tektronix and Hewlett-Packard) have started delivery to customers of high-speed oscilloscope equipment using TD trigger circuits. With the ice thus broken, it seems that this element will find its way into some types of circuits in digital computers within the next year or two, with serious application presumably somewhat further in the future.

One interesting possibility, considering the extremely small size of the working parts of TD's, together with their presumably low cost when placed in true mass production, may be in fairly sizeable memory arrays having cycle times in the very-small-fractional-microsecond range.

It seems clear that a basic factor delaying widespread planning for TD circuitry in computers is the need for developing, then promulgating understanding of, new design techniques. In my opinion, this is a more serious deterrent than some of the weaknesses reported in early circuits. Furthermore, whole new concepts must be developed for fabrication, maintenance, reliability prediction and testing, and jointure with other types of circuits.

In sum, the tunnel diode appears to offer strong advantages for very-high-speed computer application, but will probably not be an important component in actual computing machines during the next several years.

One new concept which may be of great importance in the distant future has been introduced during the past year. K. R. Shoulders of Stanford Research Institute has published (Proc. WJCC, May 1960) on a striking new idea in microelectronic technology: the vacuum tunnel tetrode. This proposal involves the fabrication (by the reversed-electron-microscope techniques of the late D. A. Buck and his associates at M.I.T.) of large arrays of point-emission-cathode tetrode vacuum tubes, interconnected only by electron streams. With typical element dimensions on the order of one

micron and packing densities on the order of 10^{11} elements per cubic inch, individual switching times would be 10^{-10} second!

Certain basic experiments have been completed by Shoulders which indicate that most of the apparently serious difficulties (environment, reproducibility, uniformity, reliability, and electron-beam micromachining) are not prohibitive. It appears that further investigation of this approach to microelectronics may produce results of sweeping importance, perhaps within five to ten years.

Let us now return to the somewhat prosaic world of transistor technology. Recent developments in very fast silicon switching diodes and transistors appear to indicate that straightforward extension of present-day computer circuits, to use the fast diode-transistor technology, may make logic element delay times on the order of 10^{-8} second (i.e., 10 nanoseconds) quite designable. It would seem that such circuits could be used to extend the speeds of some present machine components by several hundred percent.

Memory technology progress has been less dramatic. Progress continues in use of thin magnetic film techniques for developmental small memories in in the 100 nanosecond region and faster. As noted above, there may be some hope for fairly sizeable memories using tunnel diodes in the several-nanosecond speed region. I would hope that memories of thousands of words will soon be feasible using thin film techniques, and perhaps of hundreds of words using tunnel diodes.

Both of those techniques will presumably call for at least three echelons of backup memory: As at present, word-arranged magnetic core memories operating in the one-microsecond region and some tens of thousands of words in size will be available. Disc or drum auxiliary memories having large data flow rates (perhaps a million characters per second) and small-fractional-second random access times are now nearing market availability. Magnetic tape, as at present, will be a major input-output medium for the foreseeable future, and modest speed increases (to hundreds of thousands of characters per second) have already been laboratory-demonstrated.

If the remarks above relating to multiple-echelon memory convince you that new and more difficult programming problems will face us soon, we are in agreement. It is fortunate that automatic storage allocation ideas have been considered seriously by top-drawer program research people for several years, and that some workable techniques are now under development.

In the area of general-purpose machine organization in general, two basic advances which are already in operation in digital control computers will presumably achieve wide application in connection with the time-sharing of machines by real-time (i.e., prompt-results-needed) problems together with conventional data-processing applications. The first is the use of elaborate interrupt features, having powerful multiple-level-priority

control. A French-made machine, the Gamma 60, and the U. S.-built Honeywell 800 have already made these features available to customers. The beginnings of self-organization in machine-program complexes, by means of which at least major inter-connections between parts of a machine may be changed under program control to meet changing requirements or to cope with machine failures during real-time problems, having been announced by Thompson-Ramo-Wooldridge in connection with the TRW-400 computer system.

Standardization

I will confine my remarks in this area to a brief mention of the recent activities of the American Standards Association's X-3 Sectional Committee on Data Processing. This group has been charged with the responsibility of developing and disseminating American Standards for programming language and hardware, and, on behalf of the International Standards Organization, has accepted the secretariat for Technical Committee TC97 on Data Processing. As usual, the ASA will carry out this mission through enlistment of competent personnel in various professional societies and trade associations, and will in every case involving a controversy strive to achieve a true consensus rather than a mere compromise.

Projects already under way include Character Recognition; Coded Character Sets and Input/Output Media; Data Transmission; Common Programming Languages; and Problem Analysis and Definition.

This work is being carried forward on behalf of the ASA under the sponsorship of the Data Processing Group of the Office Equipment Manufacturers' Institute.

To What Extent Should Outlook Affect Planning?

In the summer of 1955, while I was at M.I.T. for a two-week study session on switching theory, I asked an old colleague, Torben Meisling (who was then in charge of Transistor circuit development at Lincoln Laboratory) whether he felt commercial transistorized large-scale machines would be avilable soon enough to affect my planning for a large-scale vacuum tube machine. At that time, the TX-2 machine was under construction at Lincoln Laboratory. Torben told me that he expected commercialization of transistor circuit technology (then being exploited in the TX-2 and similar machines) to take three to five years. We see from the viewpoint of late 1960 that he was right on target.

During these years, the technology of transistor-resistor active logic circuitry was under intensive development, together with computer system ideas planned to exploit the technology. We seem to be entering a

similar phase now with respect to tunnel diode circuitry, but the difficulties inherent in design techniques appear to be greater. Commercialization of other basically important techniques is either too far in the future to be considered seriously in present-day planning, or represents relatively modest change from 1960 techniques and hence, in my opinion, will not have substantial effect upon the computer market.

This is an exciting time in the computer business. A large, vigorously competitive computer market has grown up during the past few years in the U. S. and, to a lesser extent, in Europe. Within the past year an amazing number of new machines have been announced, and the programming support for their application by customers is under active development. In the latter connection, great emphasis is being placed upon preventing obsolescence of programs.

It is my opinion that computer facility plans developed during the next several years should look toward gradual rather than dramatic changes in machines, and in particular that future programming effort can be carried forward without fear that programs will be rendered obsolete by new machine developments.

Summary

I have reviewed for you a number of aspects of the present-day digital computer market as they affect future planning. Document Reading equipment developments were discussed briefly, along with some recent developments in source data automation, data transmission, and information storage-retrieval. Computing machine design trends were discussed in more detail, particularly with respect to new component technologies which have been widely publicized. Standardization work now being carried forward under the American Standards Association was outlined. Finally, I attempted to explain briefly my conviction that present planning for major data processing facilities can and should be carried out with reasonable confidence that future developments will not obsolete the work of the next several years.

A MANAGEMENT PERSPECTIVE ABOUT ADPS

W. H. Hill

W. Henry Hill, Director, Data Processing Systems Division, Navy Management Office, has chief staff responsibility for Navy-wide evolution of Automatic Data Processing Systems in Navy logistics and business administration. In late 1960, his concern extended into scientific and engineering uses of electronic computers. During the previous 8 years with the DCNO (Air) he developed the Navy's aircraft accounting system, made important contributions to aviation planning systems, air warfare reporting systems, and flight records and reports systems, and initiated electronic computation into those areas.

During this session, I will try to provide a perspective and concepts within which the manager not only operates but within which he may better understand what ADPS may mean to him and what he must mean to ADPS. We will attempt to portray the universal essence of management, then the nature and role of management information systems, then just how ADP fits into that in a technological perspective extending from 1940 to about 1970.

So, to begin, what is management--or at least what is a way to look at it such that every aspect of it is there, at least generically?

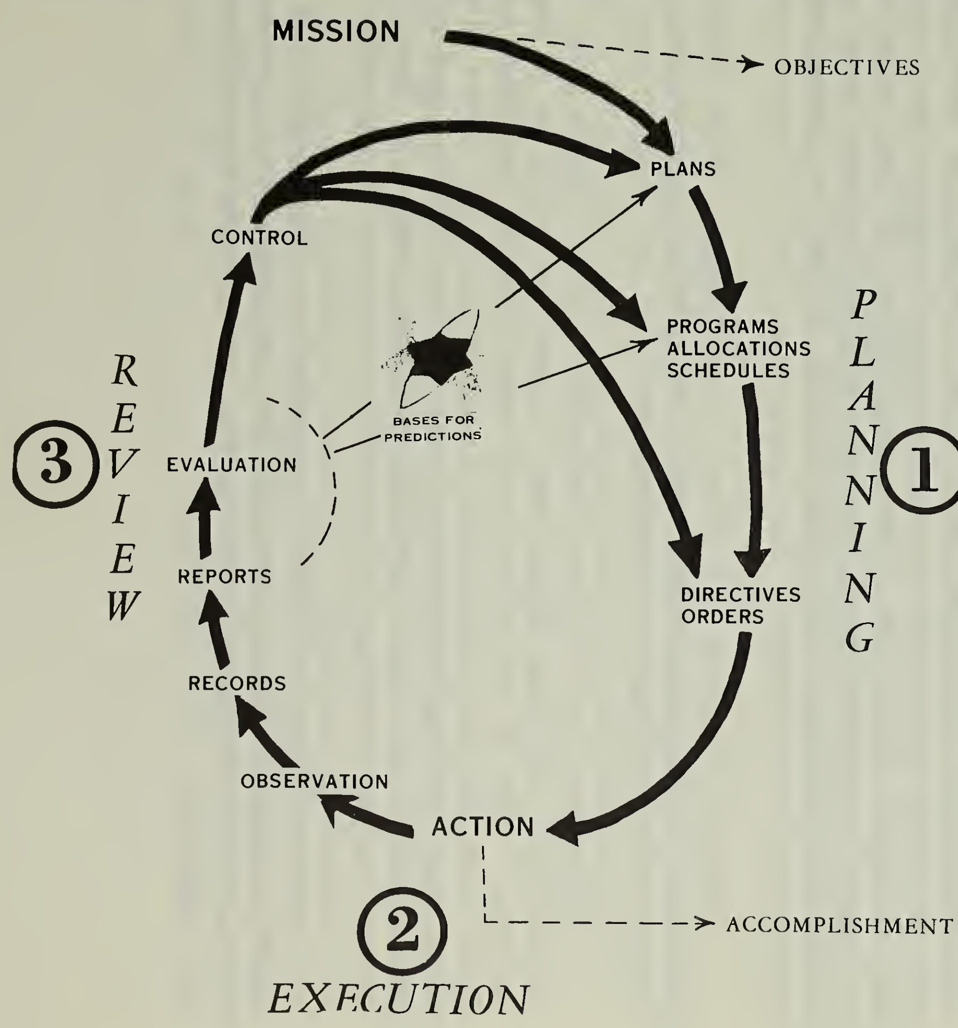
Try the accompanying chart (Figure 1 on page 231) and start with the proposition that management is a cyclical sequence of certain functions applied by various kinds and levels of management authority and responsibility to certain processes involved in certain necessary phases in the "life" of resources.

The cyclical sequence depicted by the heavy black lines and arrows we may refer to as the universal management cycle.

No rational human endeavor can fail to follow this cycle, whether the endeavor is an individual or collective one. The human being is built to operate this way--and only this way. And since this is true, any rational organization of human beings must also act this way. The two primary reasons for individuals to organize themselves are (1) to obtain greater simultaneity, i.e. get more done in any given period of time, and (2) to take better advantage of special individual proficiencies, whereof there is such a wide range throughout the human race. All this means is that a number of people have made their aggregate into a "one" that is bigger, stronger, more capable, and so on than any lesser number of them are.

Figure 1.

DATA PROCESSING FOR MANAGEMENT FUNCTIONS



MANAGEMENT -OF -		IN -
Direct, Line Responsibility	Personnel	Research and Development
- - - - - - -		Acquisition: Production, Procurement, Recruitment
Executive & Advisory Staff	Materiel	
	Facilities	Change - Physical (or Mental) Characteristics Custody Location
Supporting or Administrative Services		
	- - - - Funds - - - - -	
Coordination, Inspection, Liaison	Time Nature	Use Maintenance, Sustenance Storage Disposal, Retirement, Loss

The cycle (and the whole chart, for that matter) is just as apropos the individual concocting a martini as it is for the Department of Defense providing for national security.

The cycle is in continual motion, and it is a closed loop, but since we have to start somewhere to describe it, a good place is the compulsion step--the object--the hope--the intention--the mission--whatever you want to call it.

The next step is to PLAN, in direct response to the compulsion. This involves a number of things: general comprehension of the prevailing situation, a general description of the situation deemed most consistent with the compulsion, delineation between what will be permissible and what will not (i.e. governing principles, policies, standards, restraints, selection of a broad strategy and time-phasing out of whatever choices there are, a good idea of relative vigor and priorities, and, if necessary, a reorganization of available personnel into that which will get started (or better continue) from the present moment forward.

The next step, or management function, is to PROGRAM. This is simply an extension of the plan into more detail and in a form to which organizational units or persons can respond specifically. It lines up things to be and to happen in relation to one another, in a sequence of changes, during periods, and at points of time. Ordinarily, one first thinks of what he wants to prevail at a given point of time, then figures out what has to happen during the intervening period of time in order that it may so prevail. As used here, the word program includes both, and some other words that describe it are allocations, schedules, requirements, budgets, allowances, assignments.

The next function, in order, is to DIRECT. This communicates the plans and the programs--that which is to happen and to be--to those who are to see to it that it does and is.

Right here, if only one individual (or a single-level organization) is involved, the same people who "direct" go right on to the next step, "action." But in any multi-level organization, the DIRECT step of one level produces the "compulsion" input to the next lower level--whereupon that lower level goes around its own cycle, likewise producing compulsions--and so on, down to the lowest level, where we find the physical action whereby the compulsions are actually to be satisfied.

The next step is ACTION, presumably in conformity with the directives, which presumably convey properly the programs, which presumably are consistent with the plans, and together are presumably sufficient to satisfy the compulsion. But are they?

This question leads necessarily to the next step, OBSERVATION. Here by various sensory capacities (sight, hearing, smell, taste, touch, weight, pressure, temperature, measure, count, etc.) there is recorded in some media and language a perception of whatever is going on--not only with respect to the subjects or objects set forth in the directives, or programs, or plans, but also pertinently related features of the environment in which it is going on.

The next two steps in the cycle, RECORDS and REPORTS, we could easily have decided to include partially within the concept of "observation" and partially within the concept of "evaluation." We set them out separately, however, because of important differences when we later relate ADPS to the cycle. I think these two steps are self-explanatory.

The next function is EVALUATION. This is to order the perceptions (chronological string of observations) into useful conceptions, analysis, synthesis. Here you compare what is going on with what should be, and evaluate the differences to see what might be done about it. Here also you are accumulating intelligence--significant differences, similarities, and relationships, trends, ratios, patterns, cause and effect phenomena, rates, correlations--a body of knowledge that we hope will be (and we may call) essential valid bases for predictions. This step, EVALUATION, has three very important purposes: one, to judge the actual with the intended in order to exercise adequate control of the endeavor; two, to describe to all concerned the present situation, resources, and environment; three, to add to the library of all experience so that future plans and programs are more assured of feasibility, of greater chance of success, and of being generally more sensible. The second and third flow directly across to the plan and program steps. The first and second are inputs to the next step, "control."

CONTROL means change, as judged to be necessary, in any one or a combination of prevailing plans, programs, and directives, in order that accomplishments be bought as close as practicable to fulfillment of the objectives.

This completes the cycle.

Those management functions are exercised by people who are organized into various levels and kinds of responsibility. Level is a matter of proficiency and discipline--that is, a location in a progressive lineup of subordinate and superior degrees of authority. Kinds of responsibility are shown on the chart as four categories: the line manager, his direct executive and advisory assistance, various services which support everybody's environment and, finally--a sort of necessary nuisance--monitoring, inspection, audit, liaison.

So, then, these "management" people--within their respective subject matter cognizance and kind and level of responsibility--perform the crucial functions for the purpose of managing resources. Essentially, this amounts to the management of resources you have, managing to acquire those you need but do not have, and managing to get rid of those you have but do not need. The chart groups all resources into five: personnel, material, facilities, time, and nature. The first three are the person, something to work with or on, and a place to do it. A sixth one is also shown, money. This is a special one. Money is not really a resource; it is only a representation of a potential resource. It has two special qualities, however. Until it is used to bring about a change in the resources an enterprise has, a wide latitude remains as alternative management decisions about just what changes to use it for. The other quality is its capacity to store a concrete value in an abstract form.

And now we come to the last element of the chart--the processes involved in the "provision" of resources--the various phases in the birth, life, and death of resources.

To be truly universal, the chart should perhaps list, specifically, such things or words as "marketing and sales," "delivery," and a few others which might more clearly describe treatment of resources in private industry, governmental service agencies, social institutions, and so on. But, I believe you will find that the wording in the chart, if considered to be broad generic terminology, does indeed cover all enterprise--in fact, may include some processes in which many enterprises do not engage, e.g., research and development, or change in physical characteristics.

If, now, we can consider this a conceptual model of management, the next thing to do is to orient information systems to it--then to orient ADPS to that.

Except possibly for the "action" step in the management cycle, all the other functions of line management (and its immediate executive and advisory assistance) are essentially mental--not physical. Therefore, we organize to provide as best we can an organized complex of human minds to perform those functions. These brains will have a certain amount of intelligence in them, plus qualities of creative imagination which can add to this intelligence without recourse to an input from outside. But, the hooker is that the total knowledge required to manage adequately exceeds this.

In an enterprise of any complexity and magnitude, the management brains need information from the outside. No man lives long enough to experience personally all the experience necessary to manage properly. Ordinarily, management seldom sees or touches the resources being managed. Each brain needs an information angel. This angel we can call management information. The collective set of organized brains needs a management

information system. It would be too expensive for each brain to be augmented by a unique system.

What are some characteristics of an adequate system? Well, it certainly should serve every step in the management cycle; its very purpose is an informational service to the brains applied to those functions. Of all intelligence required, the system must provide whatever intelligence the brains do not have and cannot get by informal (non-systematic) communications. It is a "supporting service."

So, now, I submit a chart (Figure 2 on page 236) which is intended to show the essential ingredients of a complete management information system, and show it in direct relationship with the universal management cycle.

From a hardware standpoint, you can see five basic components of a complete system, and the extent to which these components and elements within them are "automatic," interconnected, and interdependent is the extent to which Automatic Data Processing Systems exist, in the broadest sense of the term--ADPS.

The five components are source data automation, data communications, data processing, data storage and retrieval, and data display (or delivery).

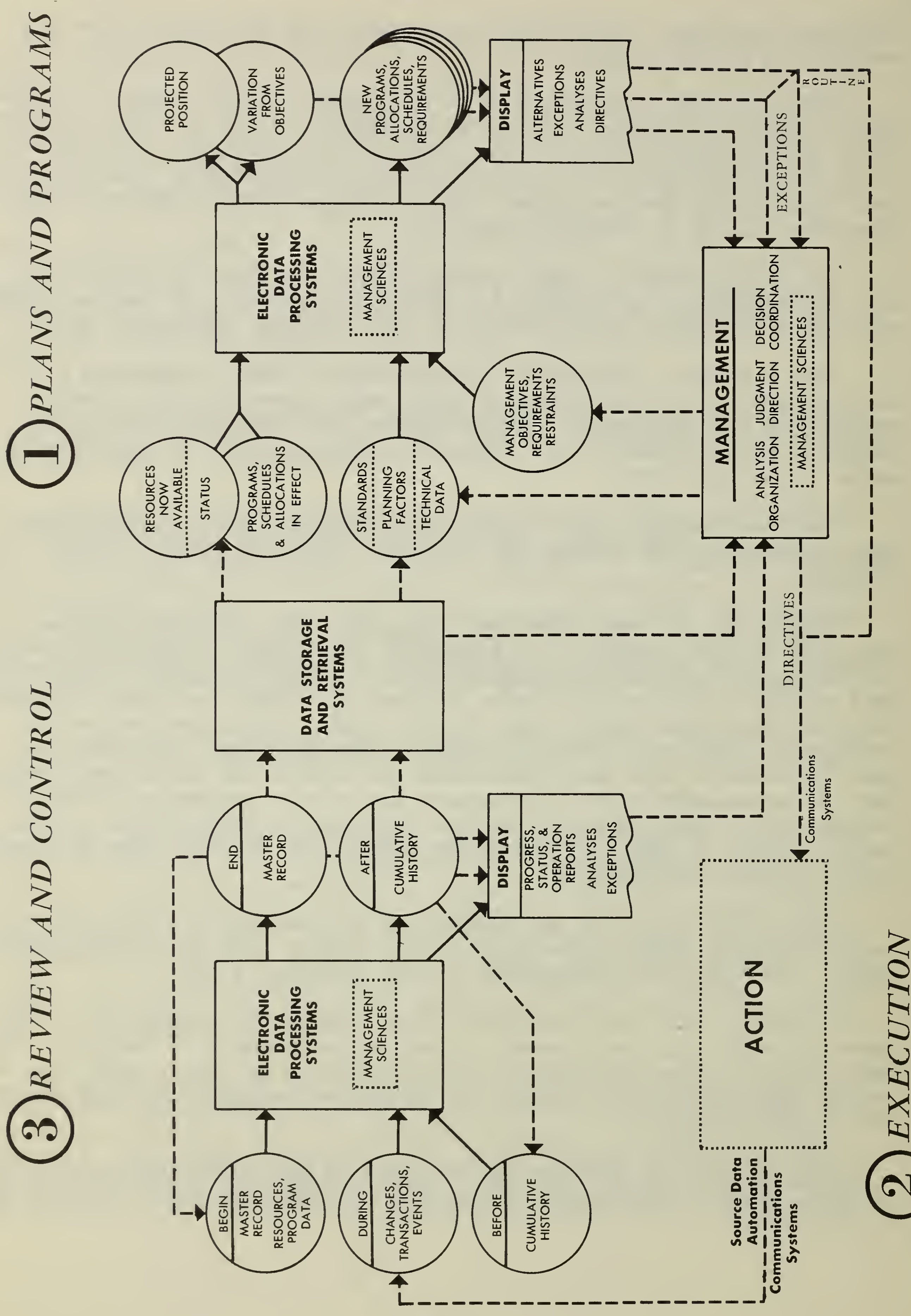
Notice the two boxes labelled "<u>electronic</u> data processing systems." This term is used because it is <u>electronic</u> technology that has occasioned all the current widespread commotion about information systems (including this course, as well as my primary occupation). Those two boxes, together with the inputs and outputs shown in the circles (which you are now encouraged to think of as reels of magnetic tape or disks), represent the only two kinds of work that practically all EDPM in the world does these days for management--both in government and private enterprise. However, to make this a true statement you must read some of the words in a broad generic sense, so that included in the output will be such items as checks, bills, notices, and the like.

If you will match the two charts--you can see that the left half of the information system corresponds to the "review" side of the management cycle. The right half corresponds to the "planning" side.

It may be said that the right half or side is the <u>production</u> of data, wherein a relatively small amount of initial information is "exploded" by application of planning factors, technical data, etc. into a great deal of detail that one finds in allocations, schedules, budgets, and requirements for specific action.

The left half or side can be described as the collection and reduction of data, wherein a great mass of raw data is summarized and translated into a useful form.

Figure 2.



1 PLANS AND PROGRAMS
2 EXECUTION
3 REVIEW AND CONTROL
BEGIN
MASTER RECORD
RESOURCES, PROGRAM DATA
DURING
CHANGES, TRANSACTIONS, EVENTS
BEFORE
CUMULATIVE HISTORY
ELECTRONIC DATA PROCESSING SYSTEMS
MANAGEMENT SCIENCES
END
MASTER RECORD
AFTER
CUMULATIVE HISTORY
DISPLAY
PROGRESS, STATUS, & OPERATION REPORTS
ANALYSES
EXCEPTIONS
DATA STORAGE AND RETRIEVAL SYSTEMS
RESOURCES NOW AVAILABLE
STATUS
PROGRAMS, SCHEDULES & ALLOCATIONS IN EFFECT
STANDARDS
PLANNING FACTORS
TECHNICAL DATA
MANAGEMENT OBJECTIVES, REQUIREMENTS RESTRAINTS
ELECTRONIC DATA PROCESSING SYSTEMS
MANAGEMENT SCIENCES
PROJECTED POSITION
VARIATION FROM OBJECTIVES
NEW PROGRAMS, ALLOCATIONS, SCHEDULES, REQUIREMENTS
DISPLAY
ALTERNATIVES
EXCEPTIONS
ANALYSES
DIRECTIVES
ROUTINE
EXCEPTIONS
MANAGEMENT
ANALYSIS JUDGMENT DECISION
ORGANIZATION DIRECTION COORDINATION
MANAGEMENT SCIENCES
DIRECTIVES
Communications Systems
ACTION
Source Data Automation
Communications Systems

With a touch of poetic license, the left side might also be termed the cyberduction of data, wherein comparisons between what is and what should be produce the adjusting actions necessary for the exercise of "control."

Now, I believe, is the time to leave this conceptual view of the entire matter and to pursue an orientation of it all with respect to time. To do this, I propose to span a period from 1940 through 1970, and to develop general characteristics for roughly every five-year period within that whole time, and to try to see what fundamental changes have taken and will take place. However, we will refer back to the somewhat timeless concepts already covered as we go along.

The 1940-1945 period produced the first electronic computer having the basic characteristics of equipment we now label EDP or ADP. But, it was a "computer," i.e. hardly anything more than what we now call the main frame, or arithmetic or logic unit of an extensive complex of equipment--the need for, feasibility of, and initial development of which was the primary characteristic of the 1950-1955 stage. The intervening period, 1945-1950, was largely one of further development of electronic computers for evaluation of mathematical formulas in support of scientific and engineering work. By 1955 there were a few--very few--EDP installations for anything other than that kind of work.

The fourth stage, 1955-1960, was a period of initial production and acquisition in quantity of the first generation of practical EDPM--a stage of earliest feasible applications, quick payoff, and pilot installing; --of EDP personnel development; and a first glimmer by management itself that something of considerable impact was here, but management was not sure just what, or what to do about it--except maybe to get some of it. Also during this stage there was much stirring (although about 5 years behind) in the way of gear peripheral to the EDPM itself; source data automation, communications, display devices, and management science adjuncts to the computers.

Currently, we are beginning the fifth stage, 1960-1965, which can be expected to have a number of characteristics. Technical people will consolidate their first experience with ADP and have available better criteria, standards, and methods to know what is needed and how to use it. Management personnel at all levels and key centers of management will develop a working familiarity with the full potential values of ADPS, particularly in conjunction with applied management sciences. There will be a striking shift in emphasis of ADP applications toward the most inherently significant use in development of plans, programs, budgets, schedules, and management control actions; with less management attention to ADPS applications to records, reports, and general substitution for clerical work. The fifth stage idea will be to augment peoples' brains, and to make management more effective.

There will develop a maturity of hardware, in three ways. It will be greatly improved in cost, size, speed, ease of use, flexibility, and other capacities.

A full range of information system hardware will be available and there will be an approach to a technical and cost balance among the various components, source data, communications, storage and retrieval, data processing, and display. Finally, this kind of hardware technology which we now label ADPE will have just about reached its upper limits of usefulness to management.

Right now, ending stage four and entering stage five, we are at a critical point of transition. We have pretty well buried the hokum and the charlatan of stage three (1950-1955). We have obtained from stage four a great deal of good and bad and different experience. We have weathered, I hope, the travails of the birth and infancy of any new technology. We knew back in 1955 that the <u>best</u> use of this technology would not be obtained until 1965-1970--that it was more likely than otherwise that for a number of years data processing would cost more with ADP than without it--but that we had no way to get to stage six without going through all the stages before that one. It seems to me, for example, that the current cost of stage four ADP, per unit of work it does, is higher right now than it ever has been and ever will be again. Just coming on the market are equipments which have perhaps ten times the capacity of equipment now installed but at perhaps two-thirds the cost. Cost of programming will decline rapidly and in the past this has been almost equal to the cost of the equipment.

But there is much to be done in the period 1960-1965 to which management itself must positively contribute; the technical people are far ahead of line management. Management must support further development and investment, and not abandon this adolescent on the verge of productive manhood simply because the kid eats more than he earns. Management must contribute to the development of a more scientific method or rationale to determine needs for management information. Management must build <u>and</u> <u>use</u> adequate staff assistance in <u>advanced</u> forms of management engineering, especially management sciences. Management must support and exercise great wisdom in fostering appropriate but non-premature standardization in the ADP field.

Hardware technicians, the producers of equipment, technical personnel in enterprises using the hardware, and many line managers will, then, provide, by the end of stage five, 1960-1965, the full <u>potential</u> (but still largely unexploited) value to management.

It will remain as the principal stage six endeavor to complete the picture by bringing management and ADPS into an optimal man/machine relationship.

Some of many differences or shifts of emphasis that may be anticipated in management and "data processing" systems between now and 1970 are:

(a) From predominant application to record-keeping and reports to a predominant use in planning, prediction, programming, and management control actions;

(b) From the kind of manager considered of higher quality due to skill in management without adequate information to a kind which will be of superior quality through a management proficiency in the design and use of adequate informational systems;

(c) From an "all or nothing" and penny-accounting type of coverage to the random spot-check or sample, the scientific estimate, and the exception;

(d) From management labor in correction and adjustment to management arts and sciences in the creative and preventive--from the reactor to the determinant;

(e) From the preponderance of total effort expended on input and storage of information, to a major concern with new and more useful informational outputs;

(f) A change in ratio such that there will be fewer advisors, interpreters, finder-outers, and other human informational servants per line manager; fewer middle line managers per top line manager, and line managers constituting a greater percentage of the total personnel force;

(g) Probably the most significant of all, from a largely "serial" or "sequential" design of organizational structures and management planning and control systems, to a largely "parallel" or "simultaneous" design.

Also, from about 1965 on, the kind of equipment referred to as "ADPE" will be "conventional" equipment. It may be expected that further significant improvement through this kind of ADPE can come only through:

(a) Better equipment combinations or use of more than one equipment, component, or circuitry at the same time;

(b) Better techniques, processes, and language in its operation;

(c) Better use of it by management.

Stage 6 of conventional ADPS can be expected also to be "Stage 3" of a different technology which, among other things, will probably involve physical properties (electro-magnetic, chemical, optical, atomic, etc.) organized in a cybernetic or perceptive-conceptive-reactive mode, including a very real capacity to "learn." Associated therewith will be a management information technology of language primarily not of words and numbers, but in "animated" shapes, symbols, patterns, colors, and other displays of analogies and simulations of organized resources, occurrences, and situations. Advanced "communications structures" can be expected to be the essence of the next most significant advance in technology, particularly in the interests of best and most quickly conveying significant information into and out of the consciousness of the human beings in the organization.